Fig 1

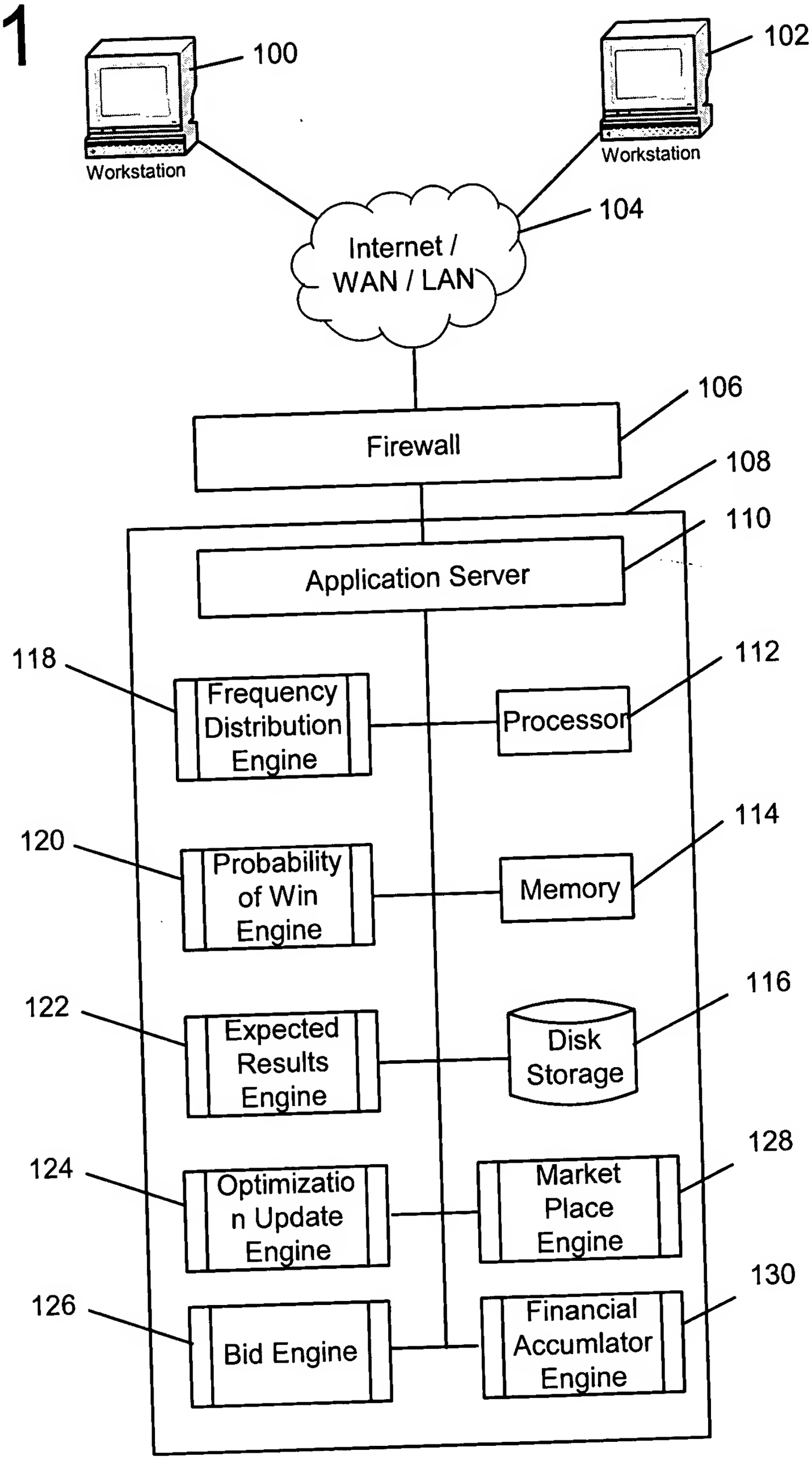

100
Workstation
102
Workstation
104
Internet / WAN / LAN
106
Firewall
108
110
Application Server
118
Frequency Distribution Engine
112
Processor
120
Probability of Win Engine
114
Memory
122
Expected Results Engine
116
Disk Storage
124
Optimizatio n Update Engine
128
Market Place Engine
126
Bid Engine
130
Financial Accumlator Engine

Fig 2

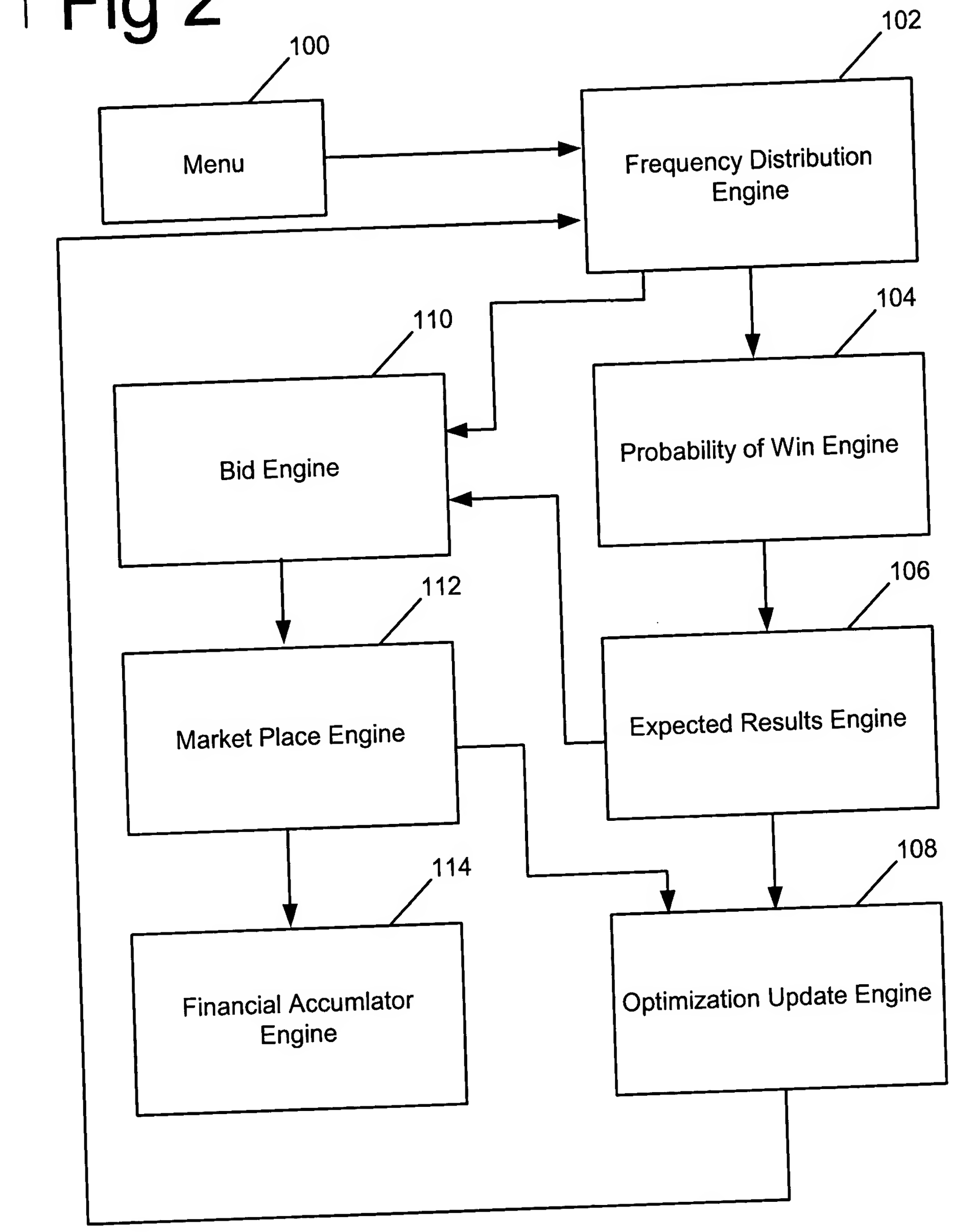

100
Menu
102
Frequency Distribution Engine
110
Bid Engine
104
Probability of Win Engine
112
Market Place Engine
106
Expected Results Engine
114
Financial Accumlator Engine
108
Optimization Update Engine

Fig 3

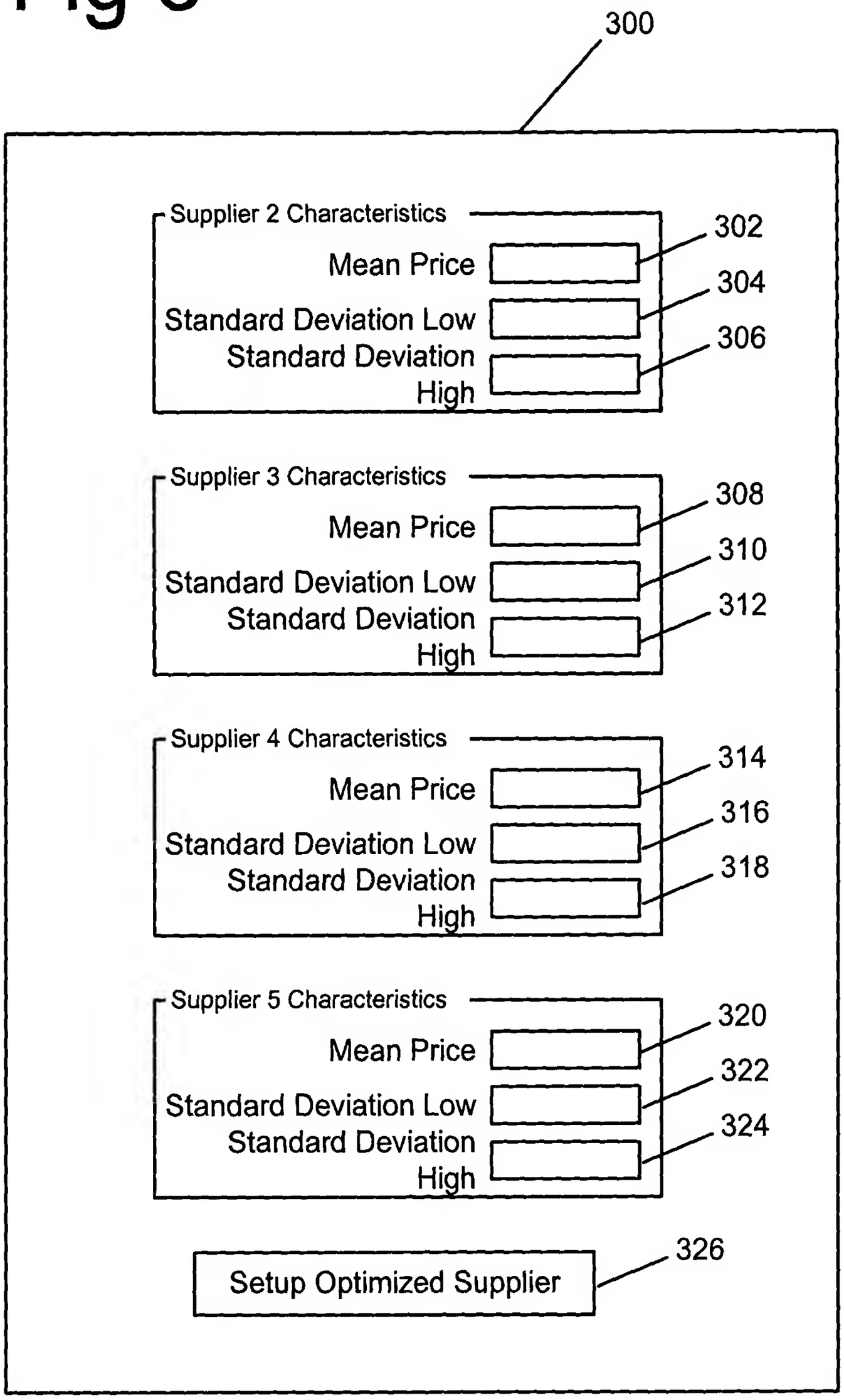

300
Supplier 2 Characteristics
Mean Price
Standard Deviation Low
Standard Deviation High
302
304
306
Supplier 3 Characteristics
Mean Price
Standard Deviation Low
Standard Deviation High
308
310
312
Supplier 4 Characteristics
Mean Price
Standard Deviation Low
Standard Deviation High
314
316
318
Supplier 5 Characteristics
Mean Price
Standard Deviation Low
Standard Deviation High
320
322
324
Setup Optimized Supplier
326

Fig 4

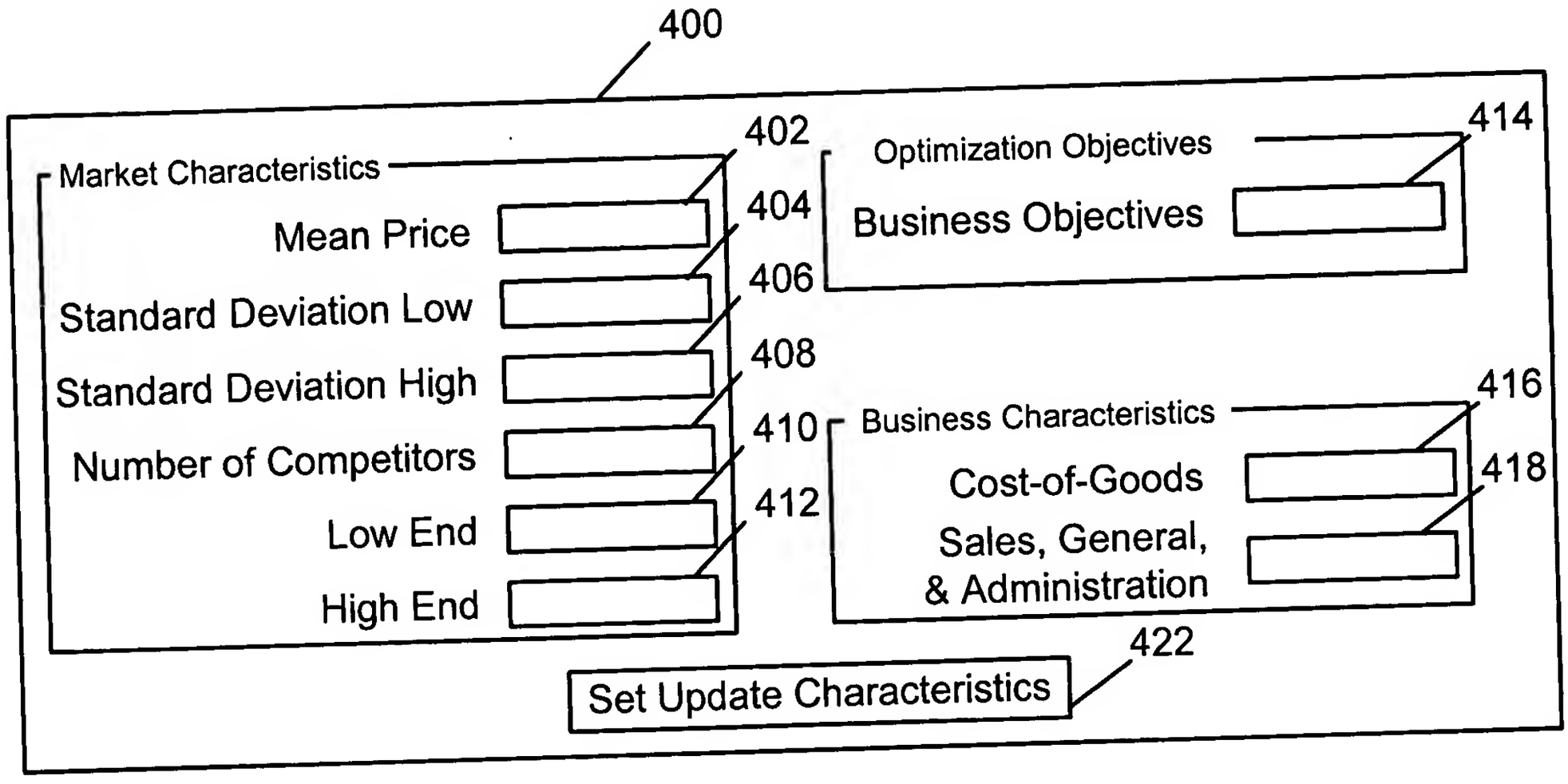

400
Market Characteristics
402
Mean Price
404
Standard Deviation Low
406
Standard Deviation High
408
Number of Competitors
410
Low End
412
High End
Optimization Objectives
414
Business Objectives
Business Characteristics
416
Cost-of-Goods
418
Sales, General, & Administration
Set Update Characteristics
422

Fig 5

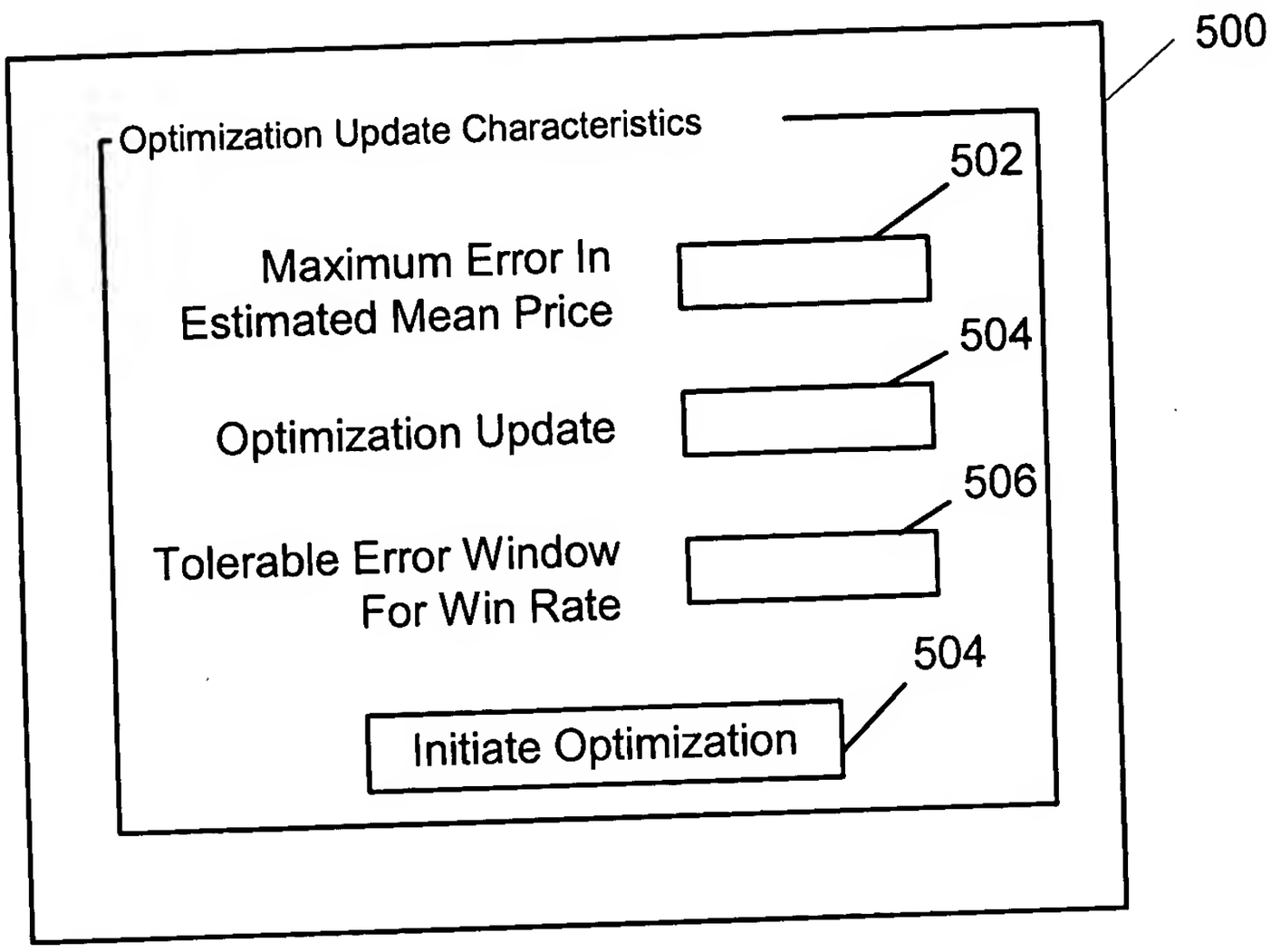

500
Optimization Update Characteristics
502
Maximum Error In Estimated Mean Price
504
Optimization Update
506
Tolerable Error Window For Win Rate
504
Initiate Optimization

Fig 6

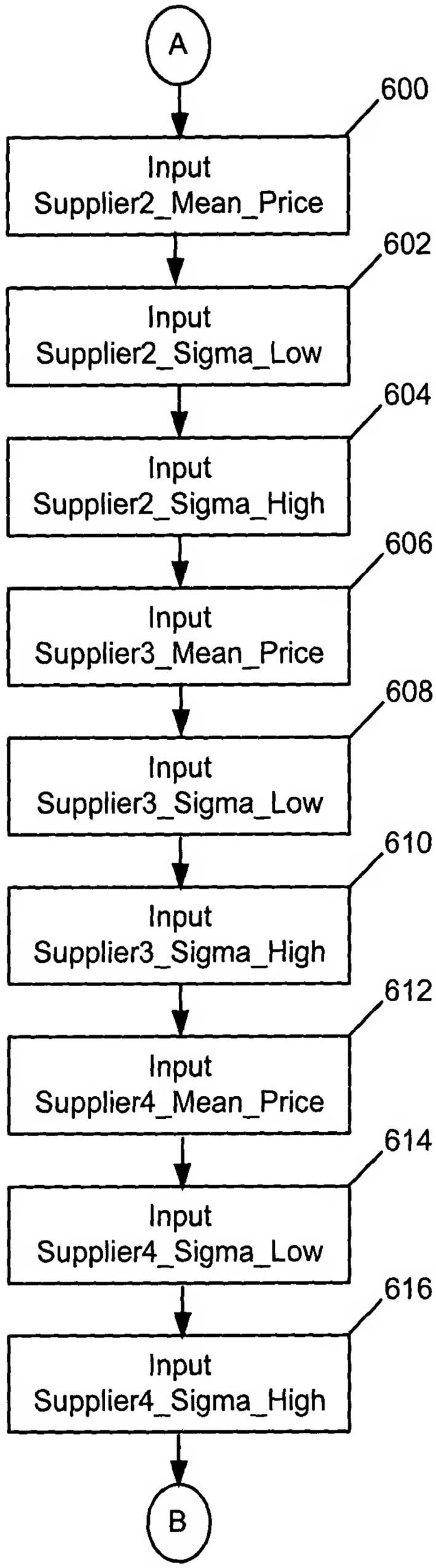

A
600
Input
Supplier2_Mean_Price
602
Input
Supplier2_Sigma_Low
604
Input
Supplier2_Sigma_High
606
Input
Supplier3_Mean_Price
608
Input
Supplier3_Sigma_Low
610
Input
Supplier3_Sigma_High
612
Input
Supplier4_Mean_Price
614
Input
Supplier4_Sigma_Low
616
Input
Supplier4_Sigma_High
B

Fig 7

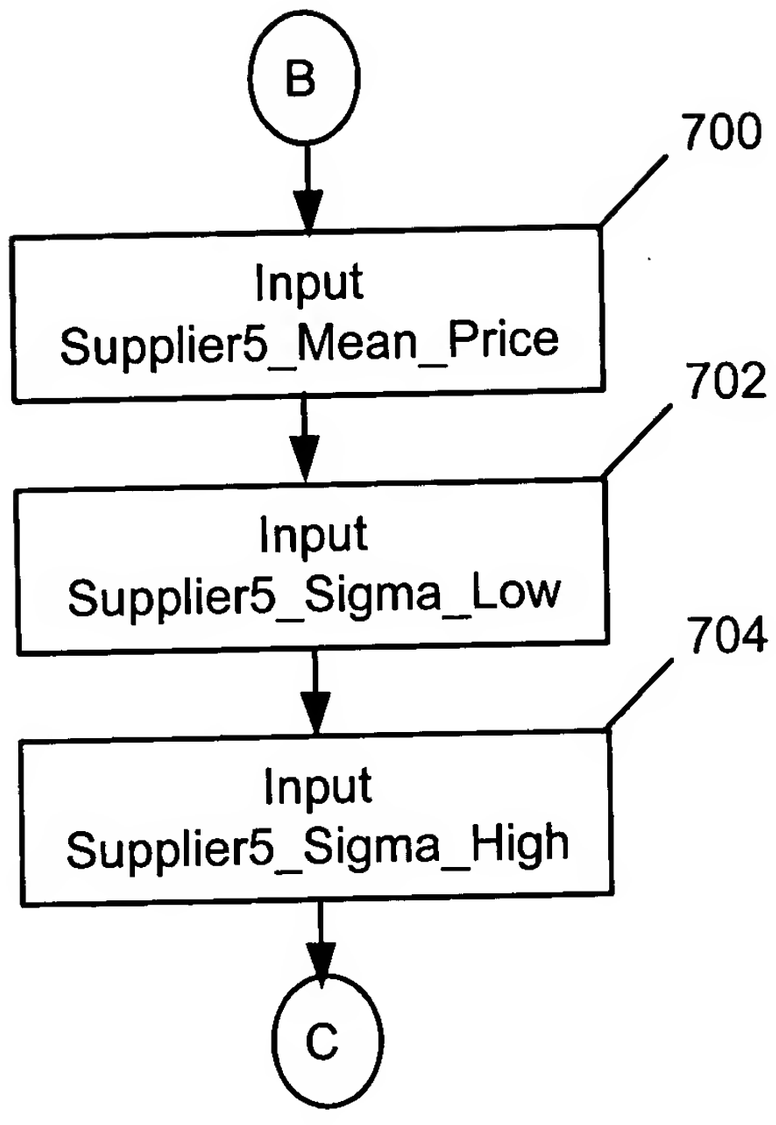

B
700
Input
Supplier5_Mean_Price
702
Input
Supplier5_Sigma_Low
704
Input
Supplier5_Sigma_High
C

Fig 8

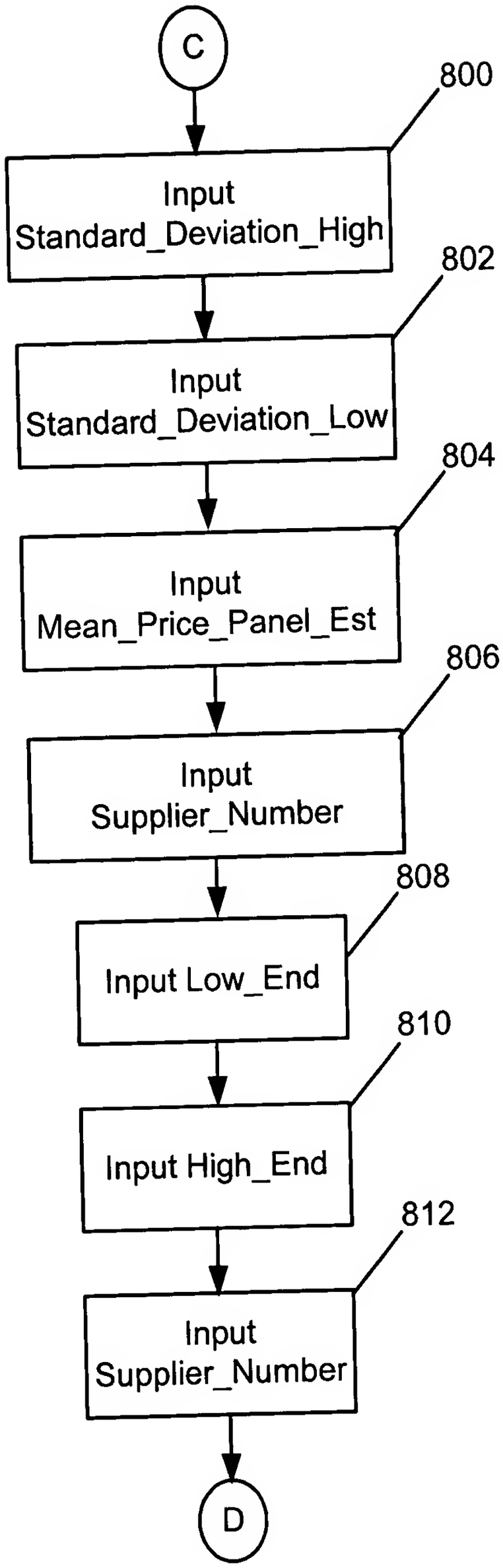

C
800
Input
Standard_Deviation_High
802
Input
Standard_Deviation_Low
804
Input
Mean_Price_Panel_Est
806
Input
Supplier_Number
808
Input Low_End
810
Input High_End
812
Input
Supplier_Number
D

Fig 9

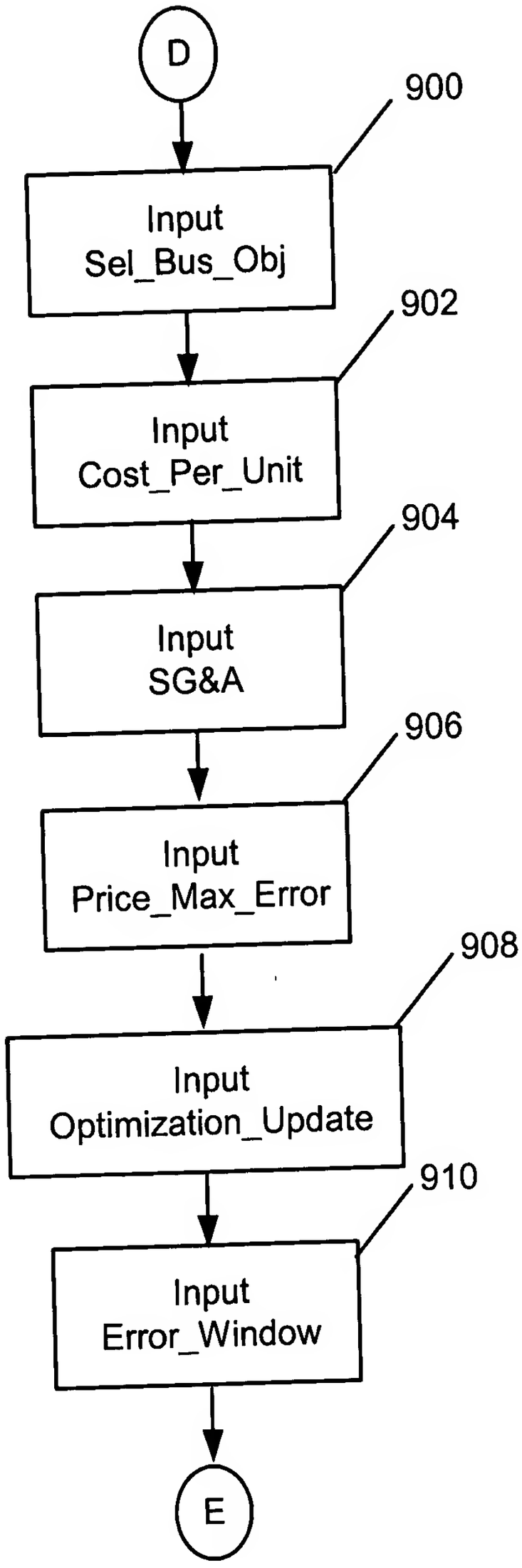

D
900
Input
Sel_Bus_Obj
902
Input
Cost_Per_Unit
904
Input
SG&A
906
Input
Price_Max_Error
908
Input
Optimization_Update
910
Input
Error_Window
E

Fig 10

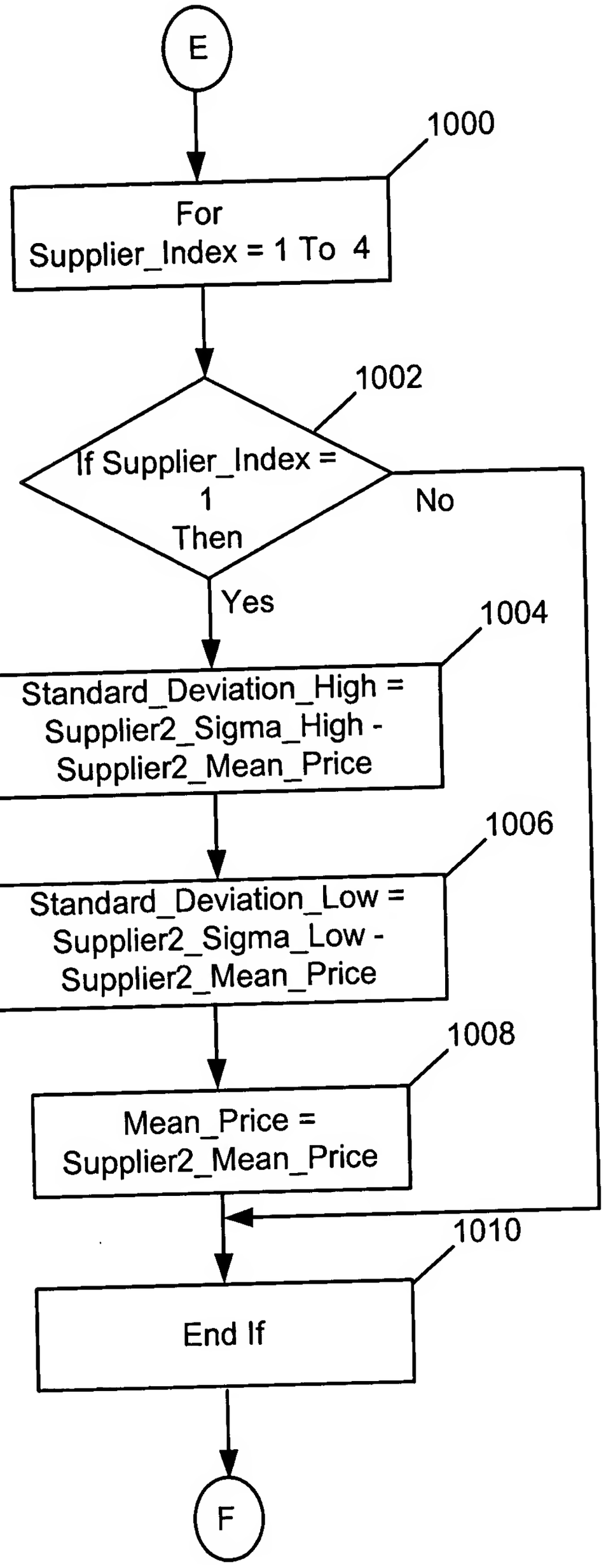

E
1000
For
Supplier_Index = 1 To 4
1002
If Supplier_Index =
1
Then
No
Yes
1004
Standard_Deviation_High =
Supplier2_Sigma_High -
Supplier2_Mean_Price
1006
Standard_Deviation_Low =
Supplier2_Sigma_Low -
Supplier2_Mean_Price
1008
Mean_Price =
Supplier2_Mean_Price
1010
End If
F

Fig 11

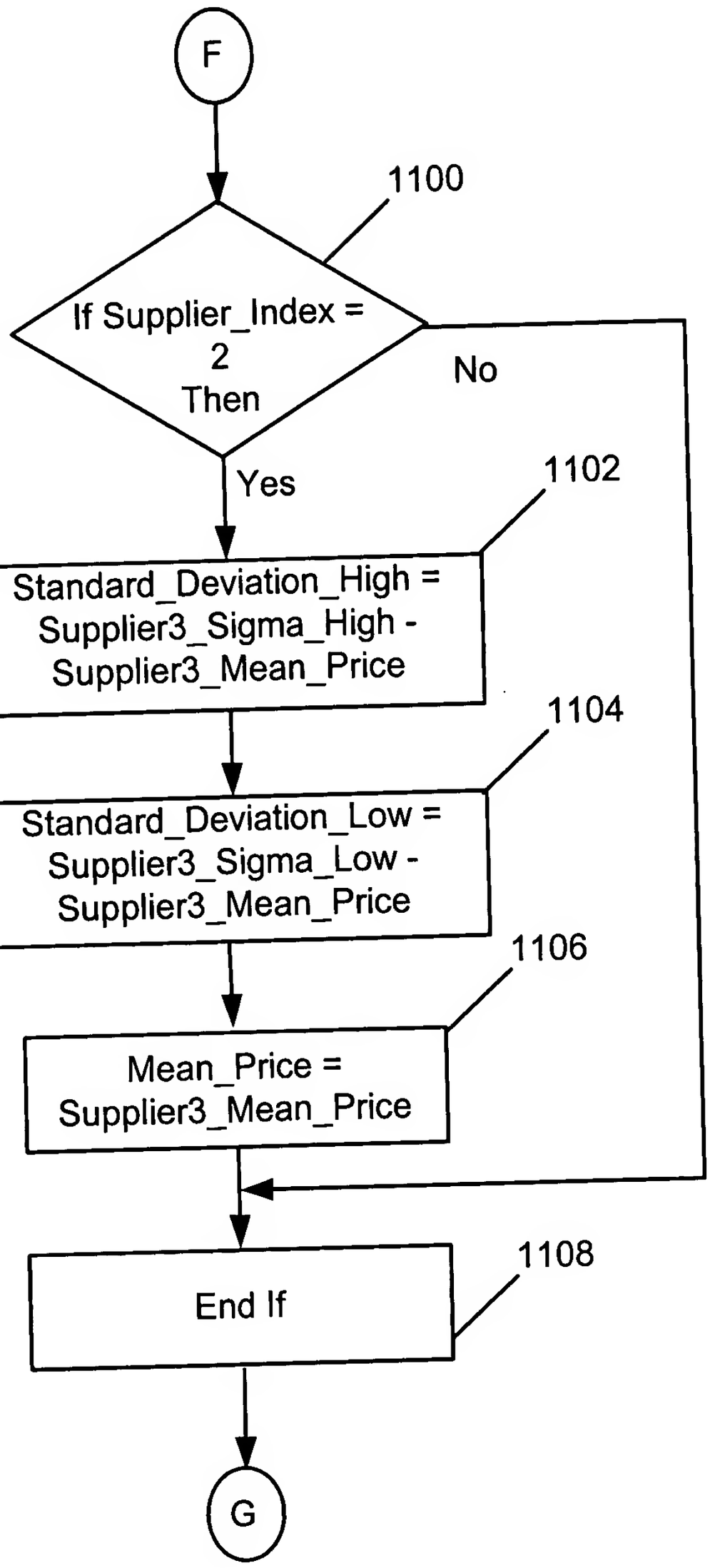

F
1100
If Supplier_Index = 2 Then
No
Yes
1102
Standard_Deviation_High = Supplier3_Sigma_High - Supplier3_Mean_Price
1104
Standard_Deviation_Low = Supplier3_Sigma_Low - Supplier3_Mean_Price
1106
Mean_Price = Supplier3_Mean_Price
End If
1108
G

Fig 12

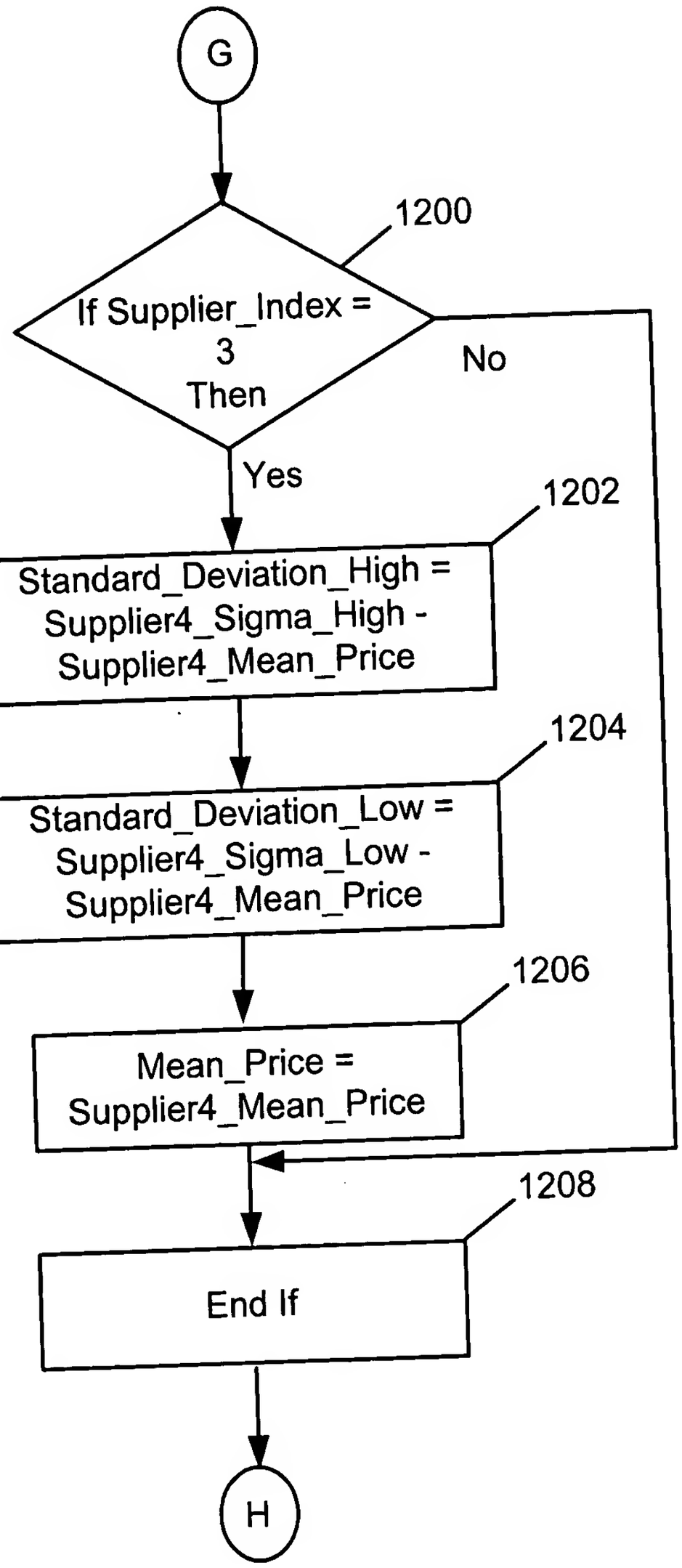

G
1200
If Supplier_Index = 3 Then
No
Yes
1202
Standard_Deviation_High = Supplier4_Sigma_High - Supplier4_Mean_Price
1204
Standard_Deviation_Low = Supplier4_Sigma_Low - Supplier4_Mean_Price
1206
Mean_Price = Supplier4_Mean_Price
1208
End If
H

Fig 13

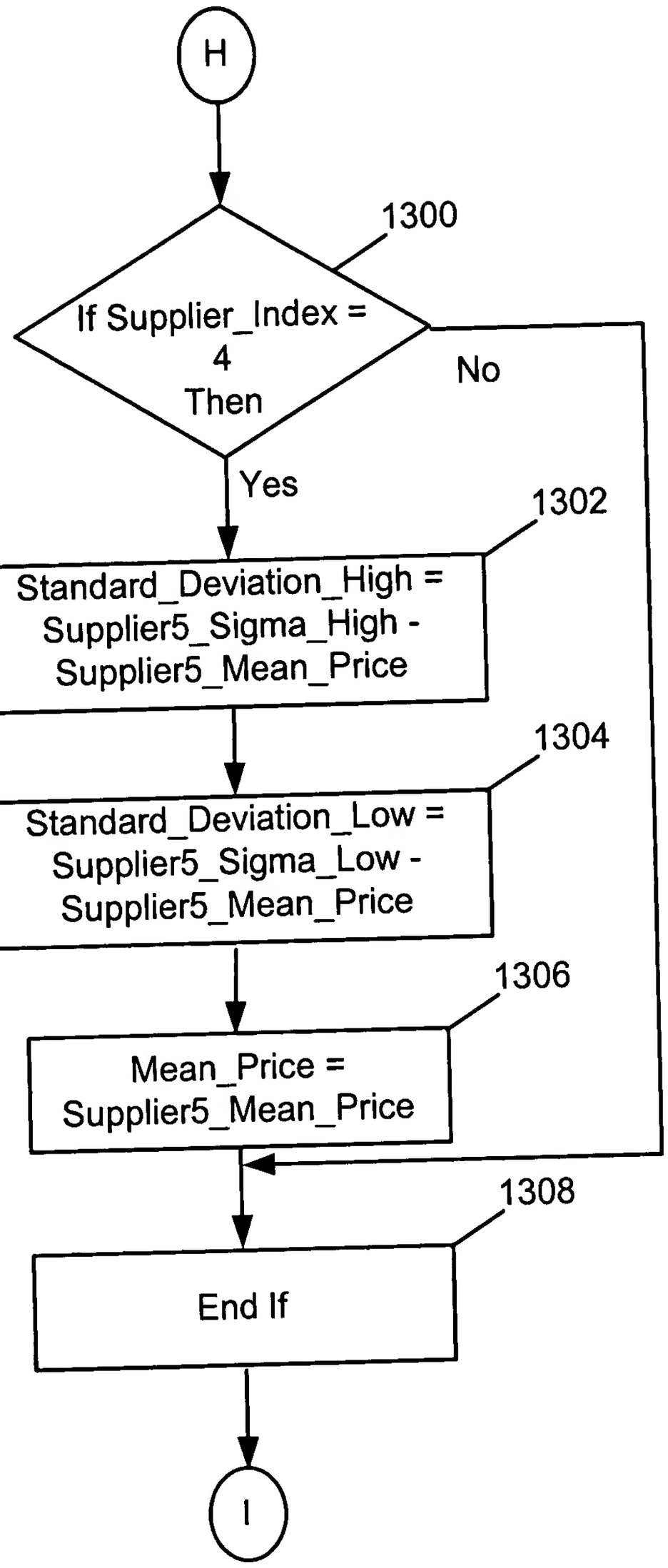

H
1300
If Supplier_Index = 4 Then
No
Yes
1302
Standard_Deviation_High = Supplier5_Sigma_High - Supplier5_Mean_Price
1304
Standard_Deviation_Low = Supplier5_Sigma_Low - Supplier5_Mean_Price
1306
Mean_Price = Supplier5_Mean_Price
1308
End If
I

Fig 14

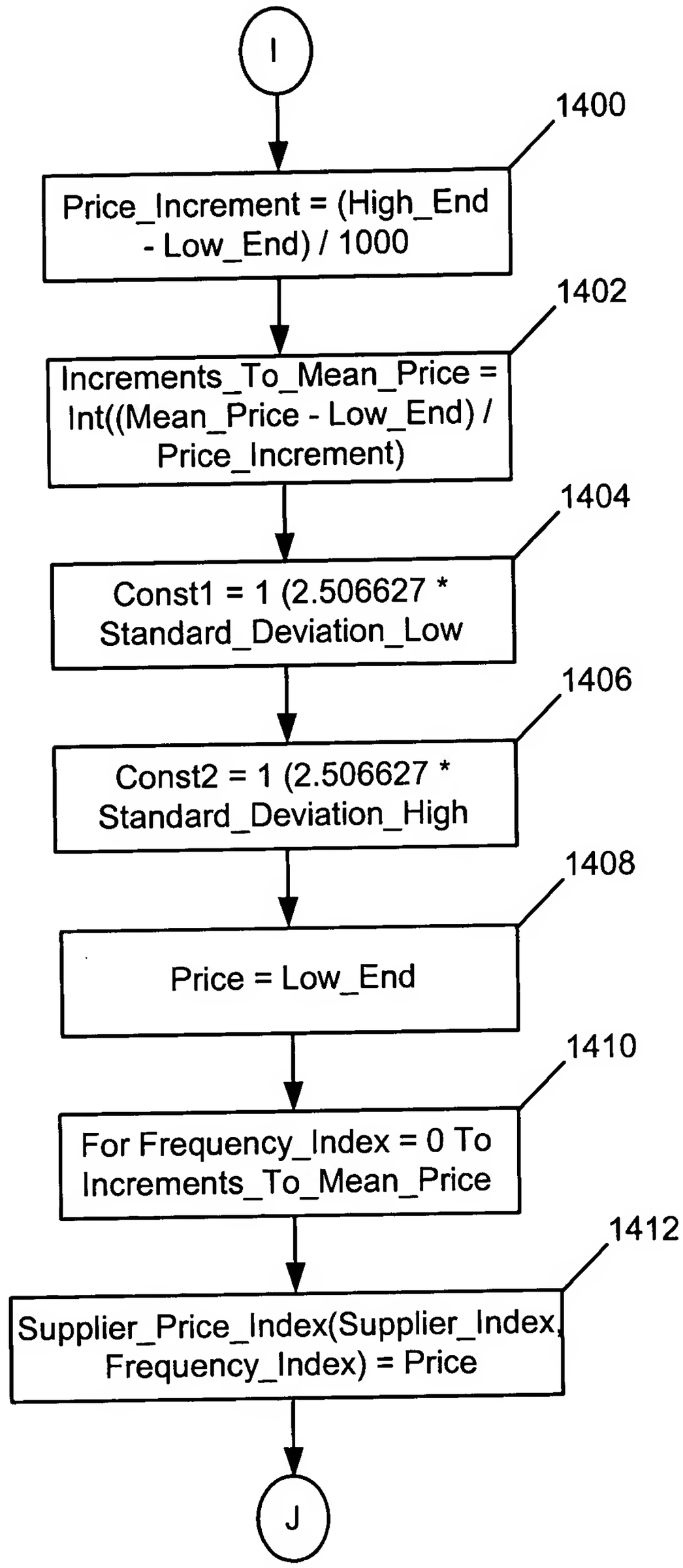

I
1400
Price_Increment = (High_End - Low_End) / 1000
1402
Increments_To_Mean_Price = Int((Mean_Price - Low_End) / Price_Increment)
1404
Const1 = 1 (2.506627 * Standard_Deviation_Low
1406
Const2 = 1 (2.506627 * Standard_Deviation_High
1408
Price = Low_End
1410
For Frequency_Index = 0 To Increments_To_Mean_Price
1412
Supplier_Price_Index(Supplier_Index, Frequency_Index) = Price
J

Fig 15

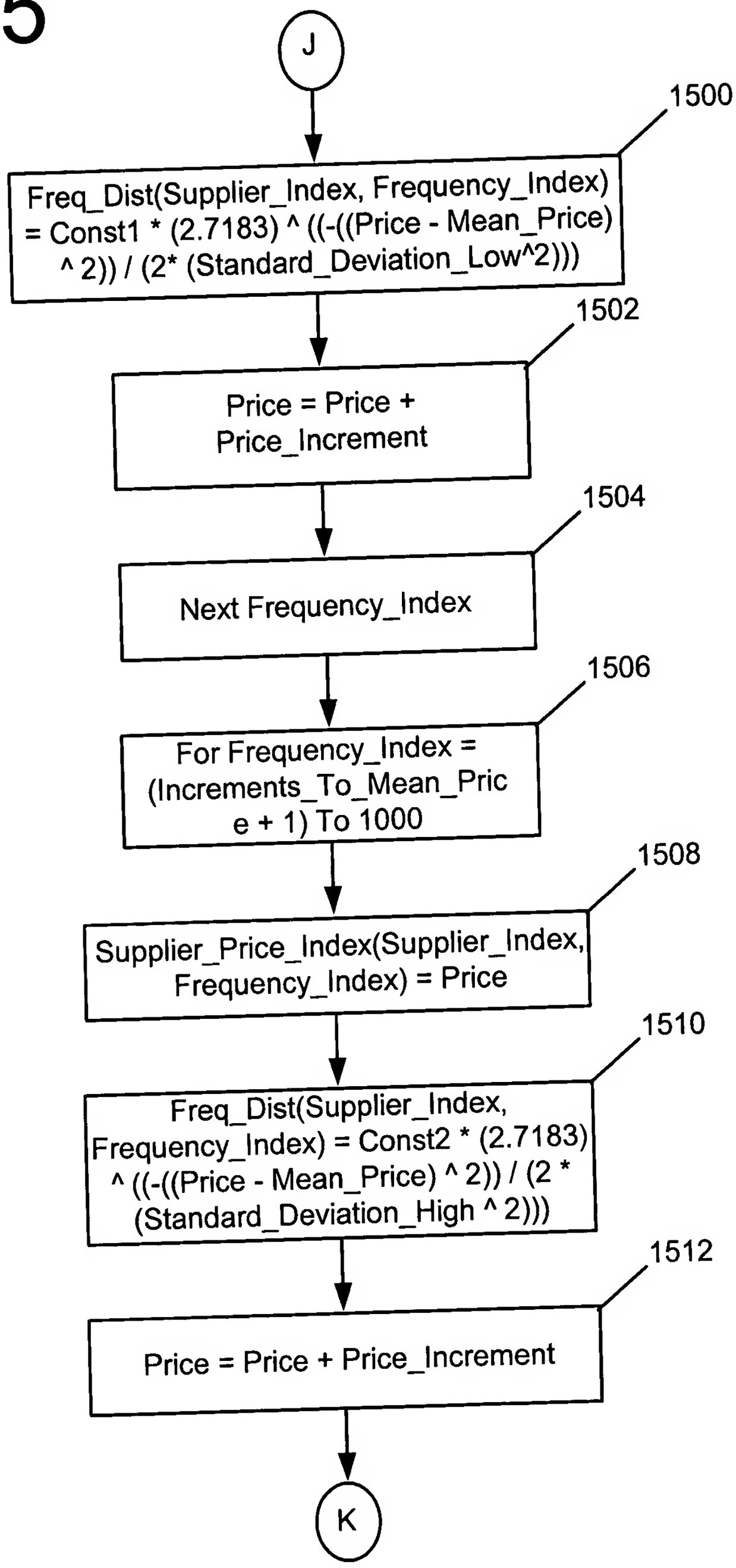

J
1500
Freq_Dist(Supplier_Index, Frequency_Index) = Const1 * (2.7183) ^ ((-((Price - Mean_Price) ^ 2)) / (2* (Standard_Deviation_Low^2)))
1502
Price = Price + Price_Increment
1504
Next Frequency_Index
1506
For Frequency_Index = (Increments_To_Mean_Price + 1) To 1000
1508
Supplier_Price_Index(Supplier_Index, Frequency_Index) = Price
1510
Freq_Dist(Supplier_Index, Frequency_Index) = Const2 * (2.7183) ^ ((-((Price - Mean_Price) ^ 2)) / (2 * (Standard_Deviation_High ^ 2)))
1512
Price = Price + Price_Increment
K

Fig 16

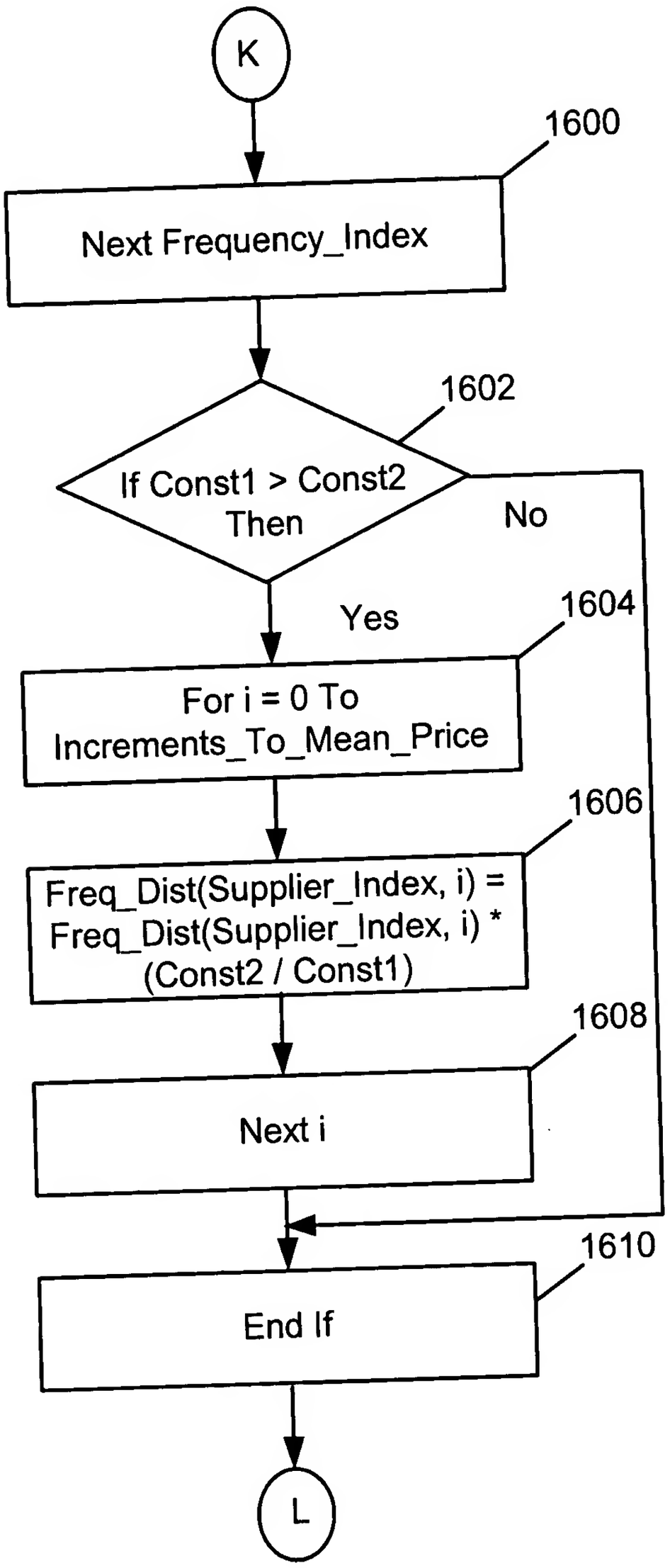

K
1600
Next Frequency_Index
1602
If Const1 > Const2
Then
No
Yes
1604
For i = 0 To
Increments_To_Mean_Price
1606
Freq_Dist(Supplier_Index, i) =
Freq_Dist(Supplier_Index, i) *
(Const2 / Const1)
1608
Next i
1610
End If
L

Fig 17

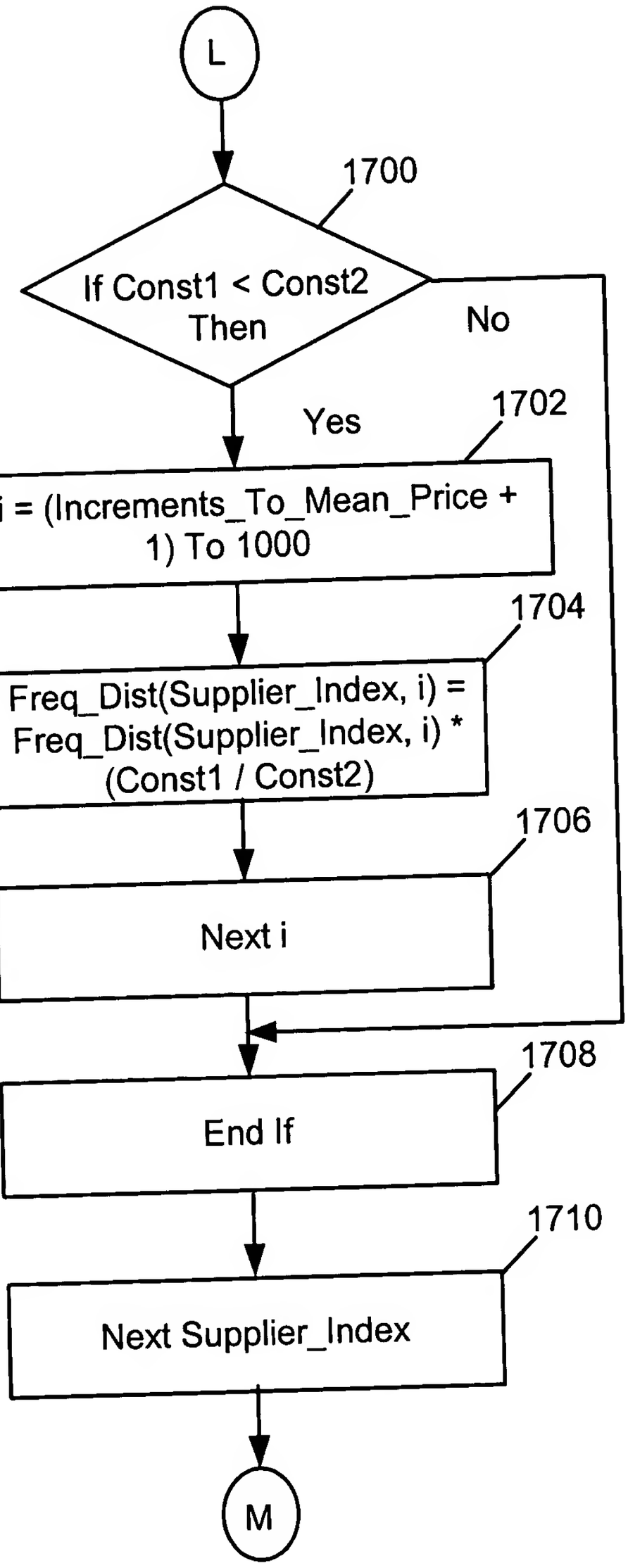

L
1700
If Const1 < Const2 Then
No
Yes
1702
For i = (Increments_To_Mean_Price + 1) To 1000
1704
Freq_Dist(Supplier_Index, i) = Freq_Dist(Supplier_Index, i) * (Const1 / Const2)
1706
Next i
1708
End If
1710
Next Supplier_Index
M

Fig 18

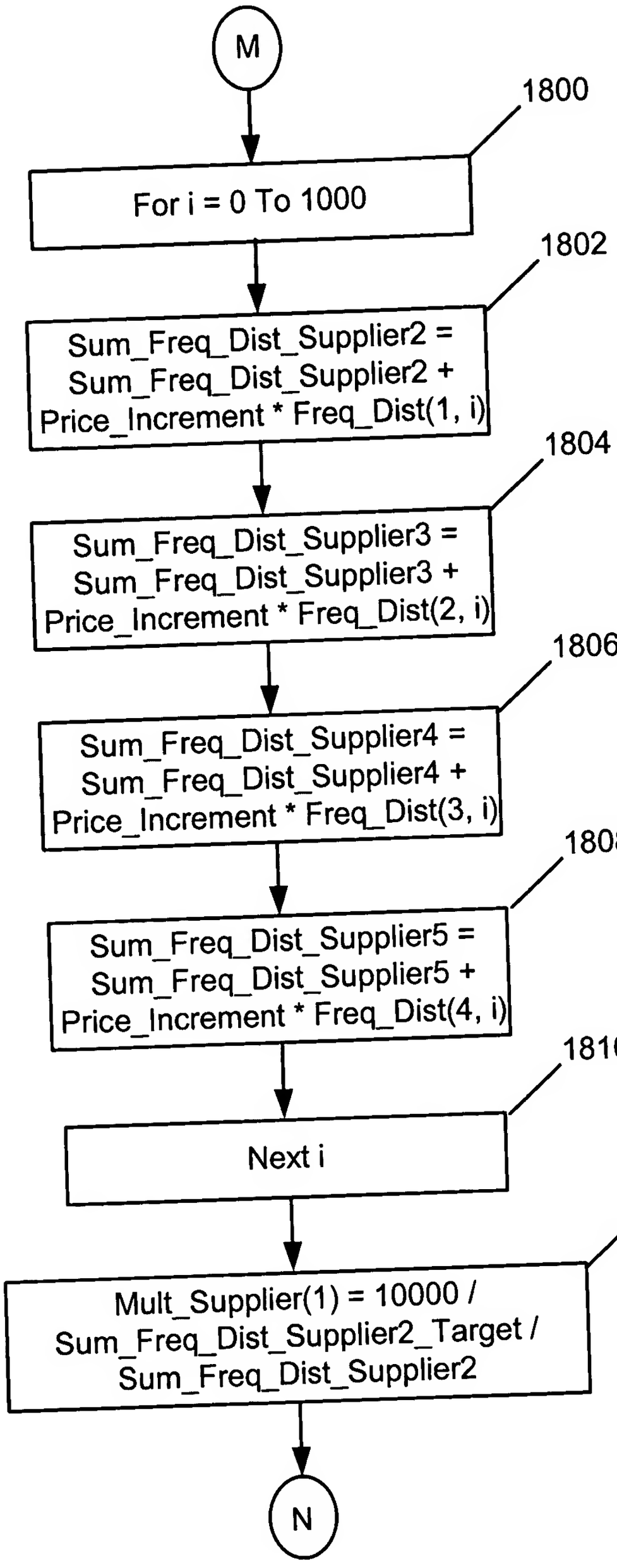

M
1800
For i = 0 To 1000
1802
Sum_Freq_Dist_Supplier2 =
Sum_Freq_Dist_Supplier2 +
Price_Increment * Freq_Dist(1, i)
1804
Sum_Freq_Dist_Supplier3 =
Sum_Freq_Dist_Supplier3 +
Price_Increment * Freq_Dist(2, i)
1806
Sum_Freq_Dist_Supplier4 =
Sum_Freq_Dist_Supplier4 +
Price_Increment * Freq_Dist(3, i)
1808
Sum_Freq_Dist_Supplier5 =
Sum_Freq_Dist_Supplier5 +
Price_Increment * Freq_Dist(4, i)
1810
Next i
1802
Mult_Supplier(1) = 10000 /
Sum_Freq_Dist_Supplier2_Target /
Sum_Freq_Dist_Supplier2
N

Fig 19

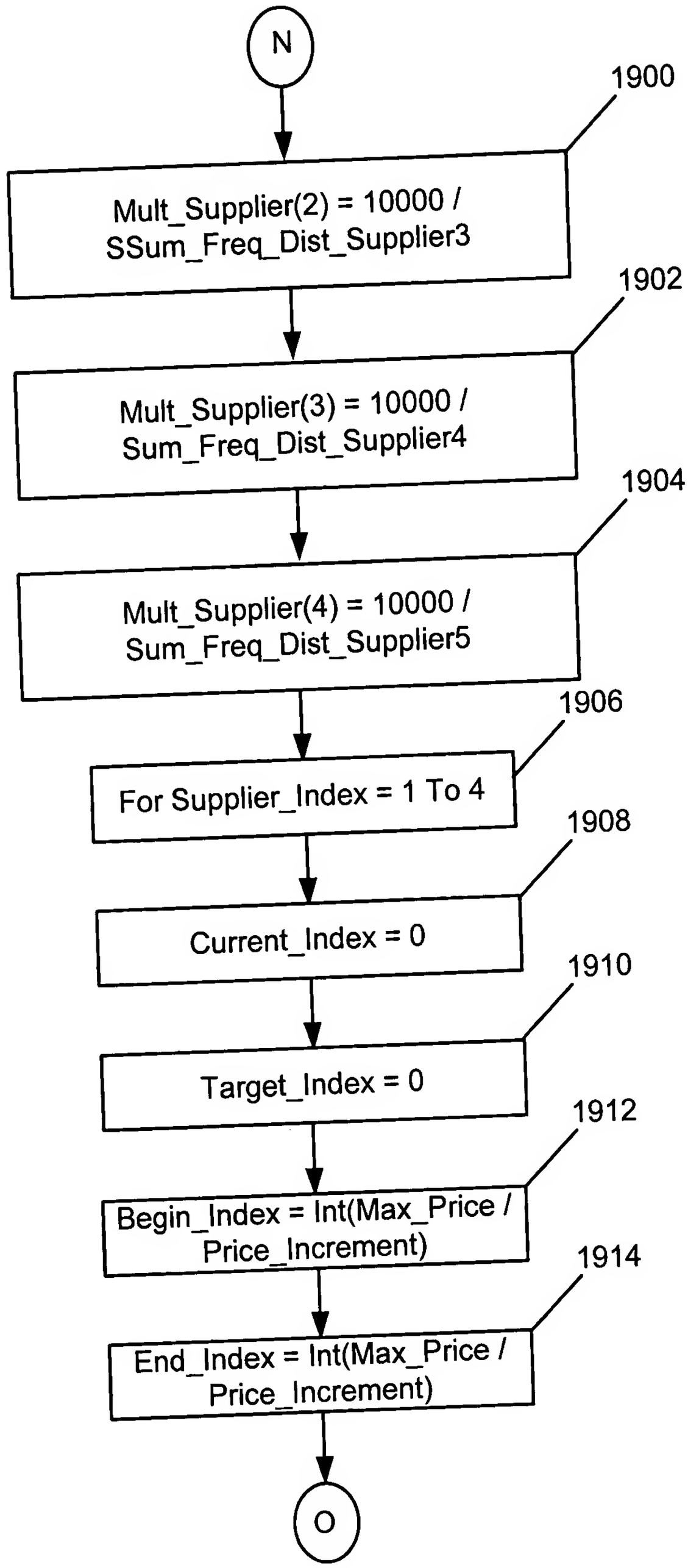

N
1900
Mult_Supplier(2) = 10000 / SSum_Freq_Dist_Supplier3
1902
Mult_Supplier(3) = 10000 / Sum_Freq_Dist_Supplier4
1904
Mult_Supplier(4) = 10000 / Sum_Freq_Dist_Supplier5
1906
For Supplier_Index = 1 To 4
1908
Current_Index = 0
1910
Target_Index = 0
1912
Begin_Index = Int(Max_Price / Price_Increment)
1914
End_Index = Int(Max_Price / Price_Increment)
O

Fig 20

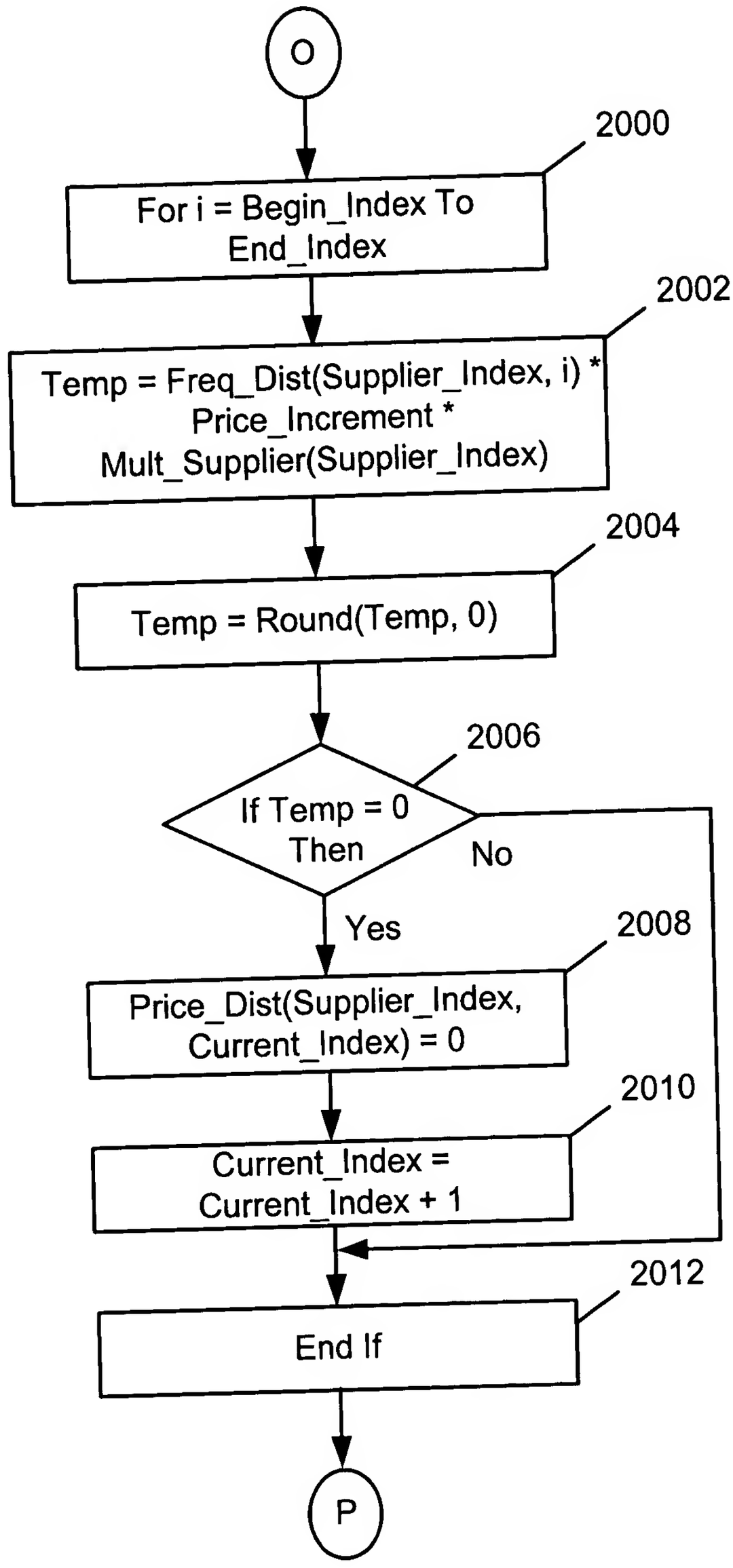

O
2000
For i = Begin_Index To End_Index
2002
Temp = Freq_Dist(Supplier_Index, i) * Price_Increment * Mult_Supplier(Supplier_Index)
2004
Temp = Round(Temp, 0)
2006
If Temp = 0 Then
No
Yes
2008
Price_Dist(Supplier_Index, Current_Index) = 0
2010
Current_Index = Current_Index + 1
2012
End If
P

Fig 21

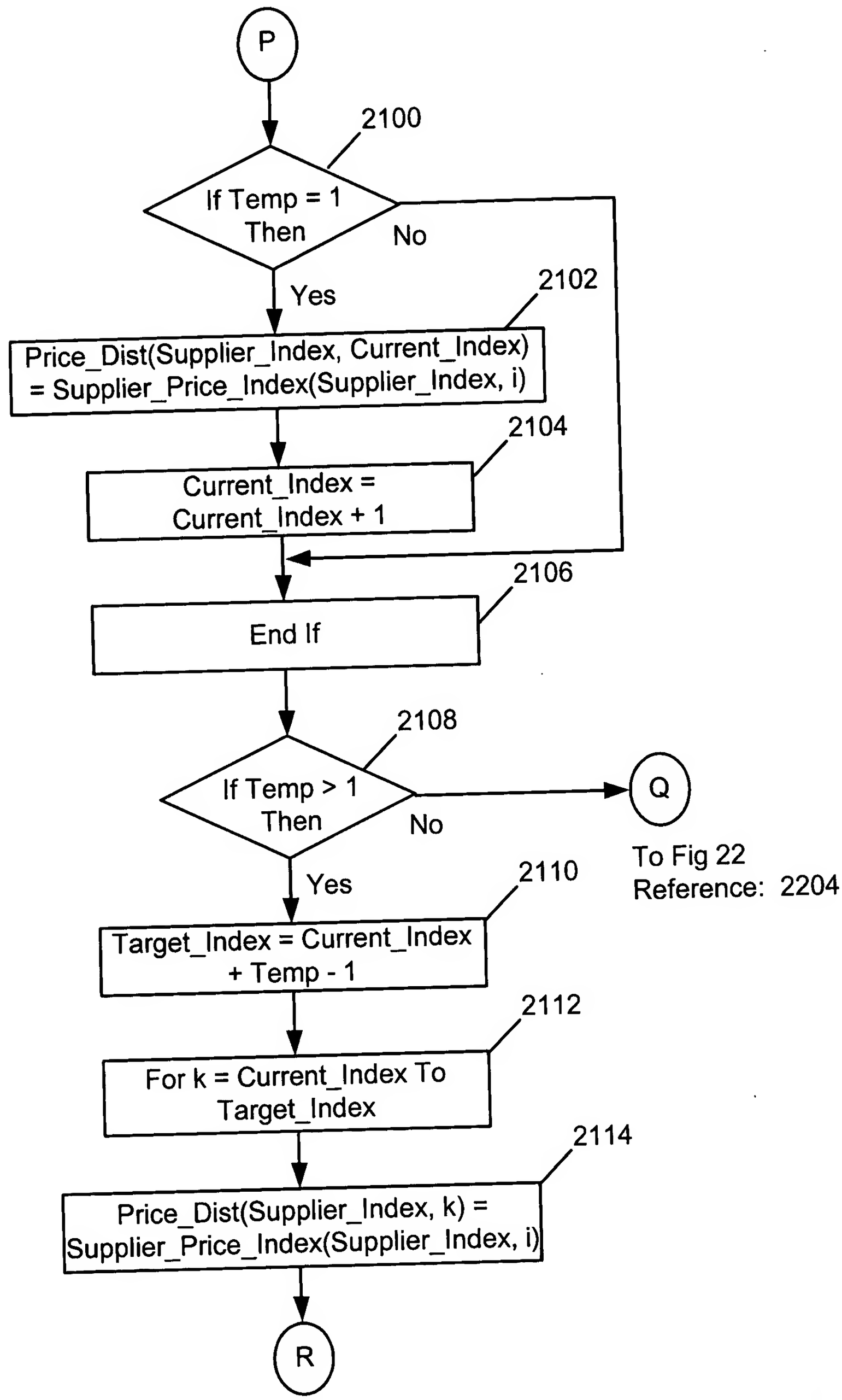

P
2100
If Temp = 1
Then
No
Yes
2102
Price_Dist(Supplier_Index, Current_Index)
= Supplier_Price_Index(Supplier_Index, i)
2104
Current_Index =
Current_Index + 1
2106
End If
2108
If Temp > 1
Then
No
Q
To Fig 22
Reference: 2204
Yes
2110
Target_Index = Current_Index
+ Temp - 1
2112
For k = Current_Index To
Target_Index
2114
Price_Dist(Supplier_Index, k) =
Supplier_Price_Index(Supplier_Index, i)
R

Fig 22

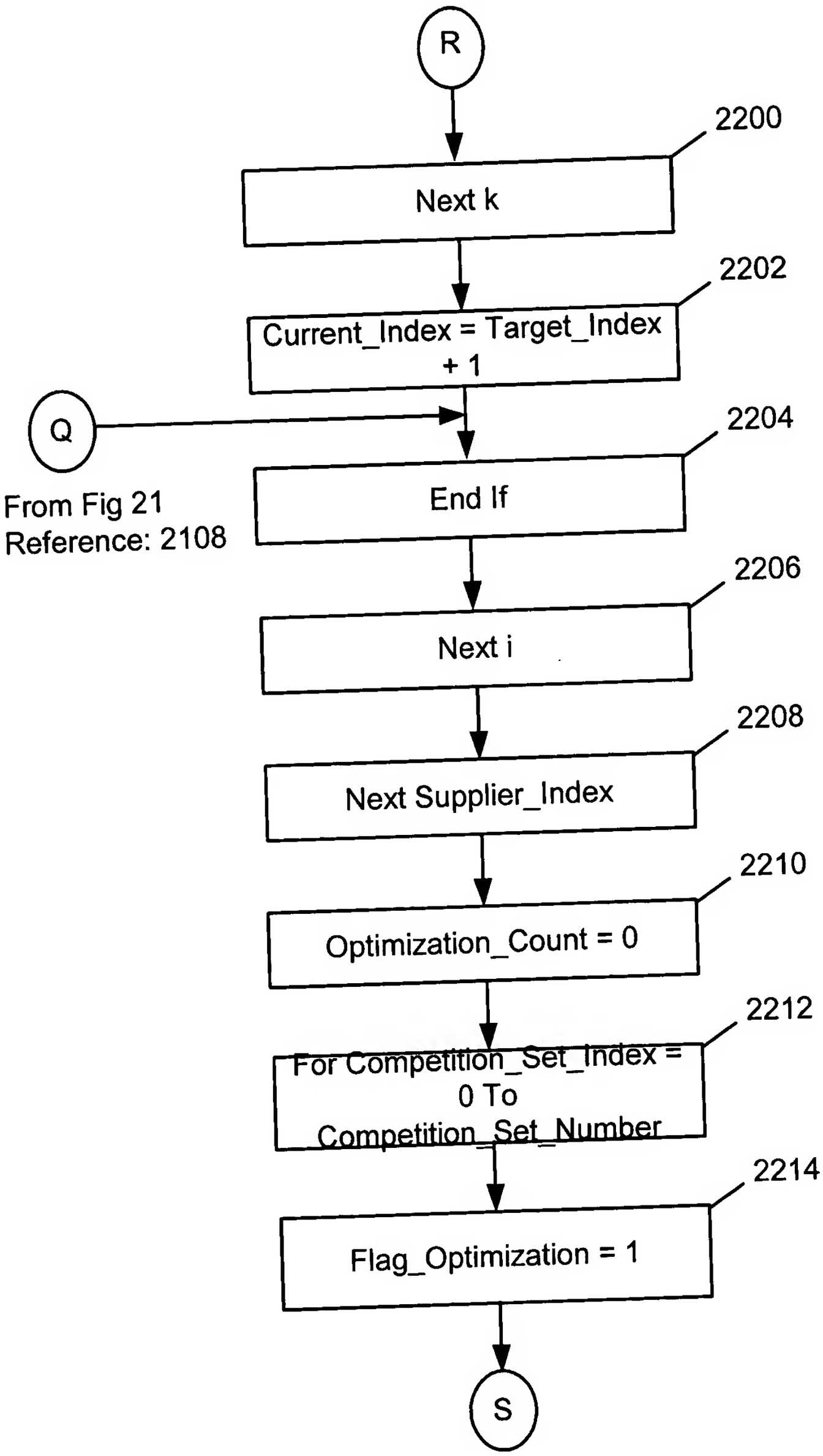

R
2200
Next k
2202
Current_Index = Target_Index + 1
Q
From Fig 21
Reference: 2108
2204
End If
2206
Next i
2208
Next Supplier_Index
2210
Optimization_Count = 0
2212
For Competition_Set_Index = 0 To Competition_Set_Number
2214
Flag_Optimization = 1
S

Fig 23

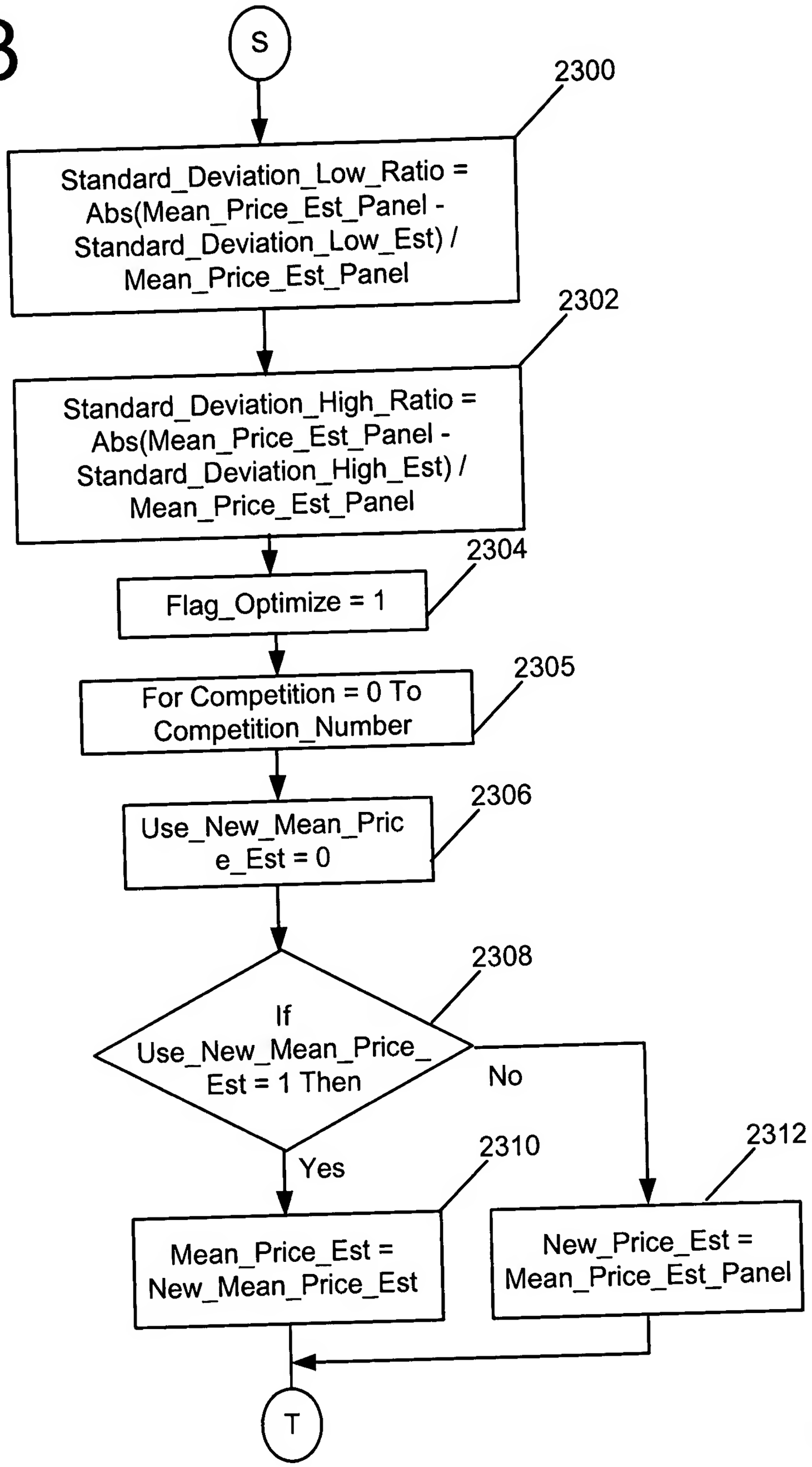

S
2300
Standard_Deviation_Low_Ratio = Abs(Mean_Price_Est_Panel - Standard_Deviation_Low_Est) / Mean_Price_Est_Panel
2302
Standard_Deviation_High_Ratio = Abs(Mean_Price_Est_Panel - Standard_Deviation_High_Est) / Mean_Price_Est_Panel
2304
Flag_Optimize = 1
2305
For Competition = 0 To Competition_Number
2306
Use_New_Mean_Pric e_Est = 0
2308
If Use_New_Mean_Price_ Est = 1 Then
No
Yes
2310
Mean_Price_Est = New_Mean_Price_Est
2312
New_Price_Est = Mean_Price_Est_Panel
T

Fig 24

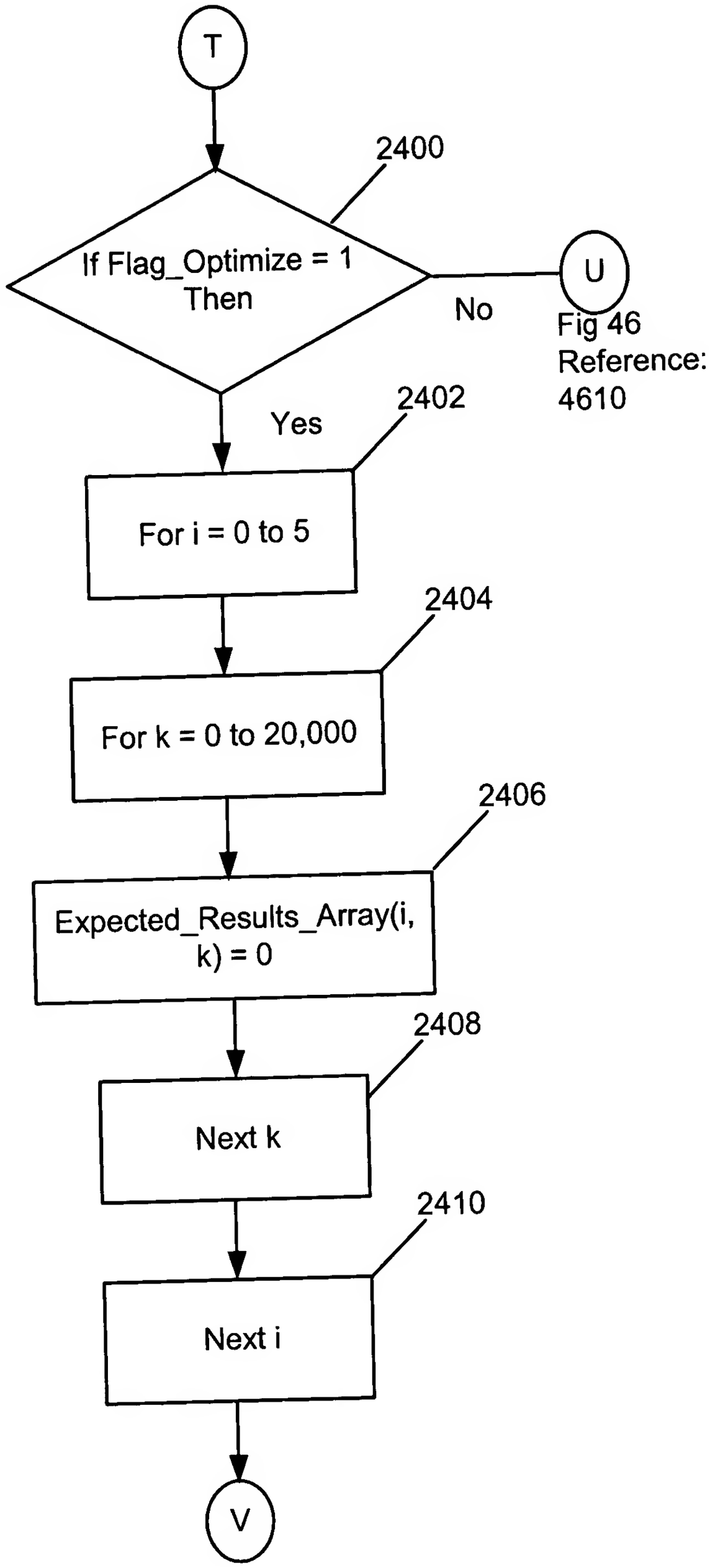

T
2400
If Flag_Optimize = 1
Then
No
U
Fig 46
Reference:
4610
Yes
2402
For i = 0 to 5
2404
For k = 0 to 20,000
2406
Expected_Results_Array(i,
k) = 0
2408
Next k
2410
Next i
V

Fig 25

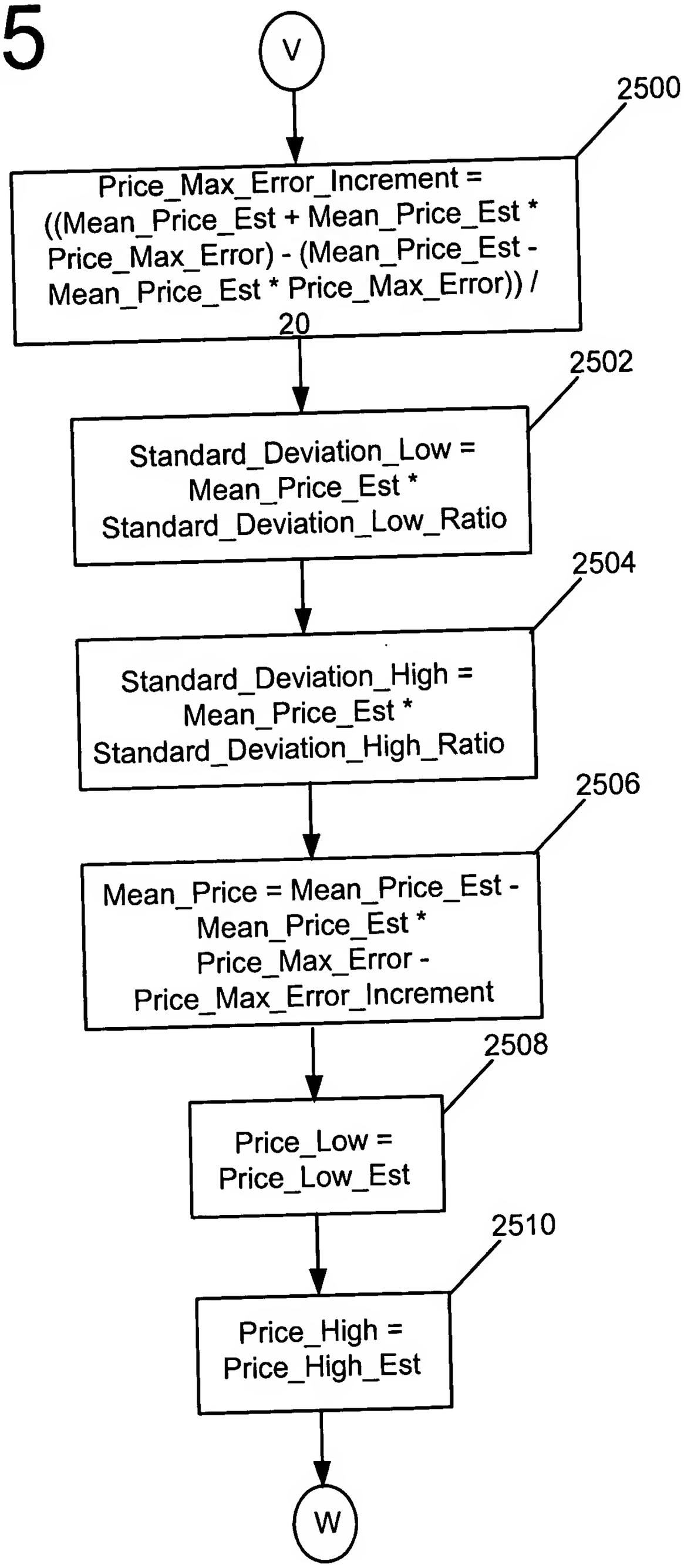

V
2500
Price_Max_Error_Increment = ((Mean_Price_Est + Mean_Price_Est * Price_Max_Error) - (Mean_Price_Est - Mean_Price_Est * Price_Max_Error)) / 20
2502
Standard_Deviation_Low = Mean_Price_Est * Standard_Deviation_Low_Ratio
2504
Standard_Deviation_High = Mean_Price_Est * Standard_Deviation_High_Ratio
2506
Mean_Price = Mean_Price_Est - Mean_Price_Est * Price_Max_Error - Price_Max_Error_Increment
2508
Price_Low = Price_Low_Est
2510
Price_High = Price_High_Est
W

Fig 26

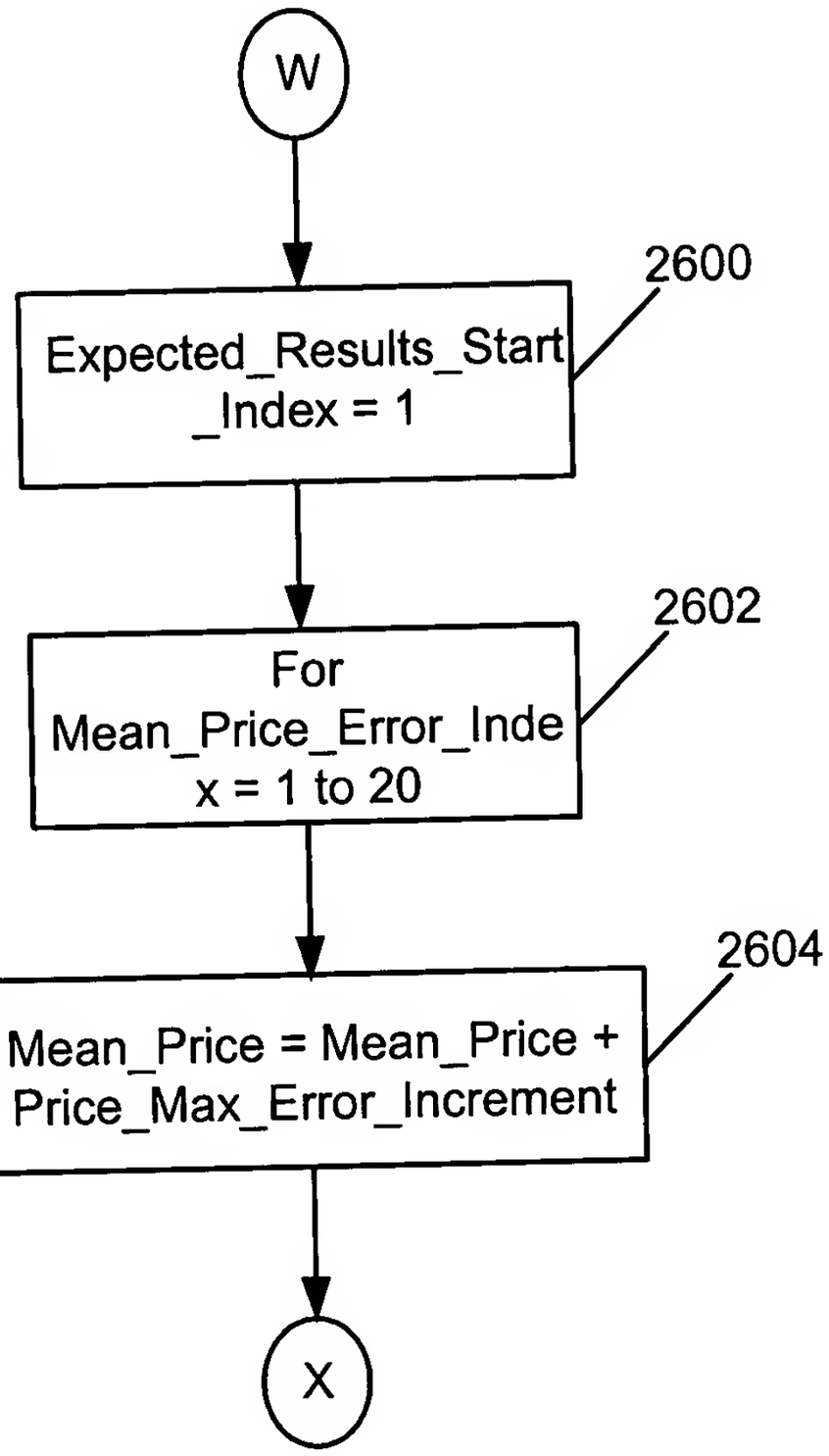

W
2600
Expected_Results_Start
_Index = 1
2602
For
Mean_Price_Error_Inde
x = 1 to 20
2604
Mean_Price = Mean_Price +
Price_Max_Error_Increment
X

Fig 27

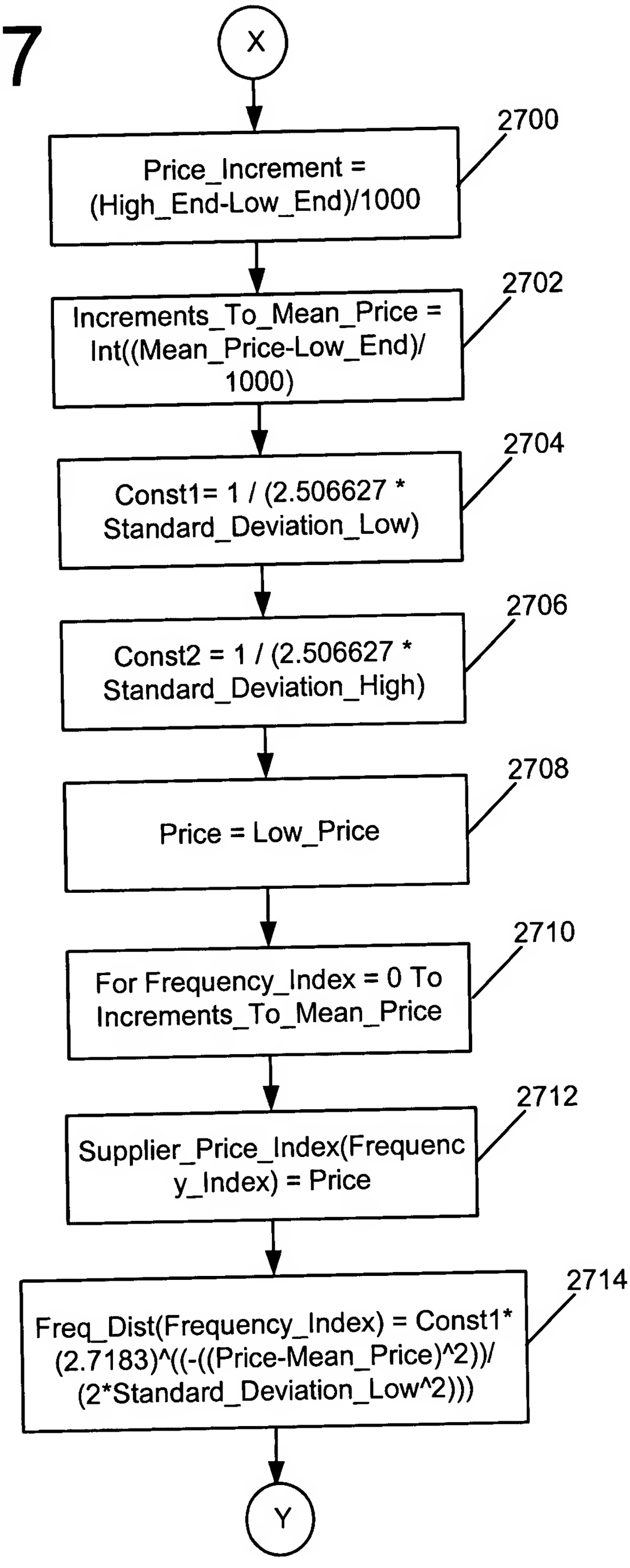

X
Price_Increment = (High_End-Low_End)/1000
2700
Increments_To_Mean_Price = Int((Mean_Price-Low_End)/ 1000)
2702
Const1= 1 / (2.506627 * Standard_Deviation_Low)
2704
Const2 = 1 / (2.506627 * Standard_Deviation_High)
2706
Price = Low_Price
2708
For Frequency_Index = 0 To Increments_To_Mean_Price
2710
Supplier_Price_Index(Frequency_Index) = Price
2712
Freq_Dist(Frequency_Index) = Const1* (2.7183)^((-((Price-Mean_Price)^2))/ (2*Standard_Deviation_Low^2)))
2714
Y

Fig 28

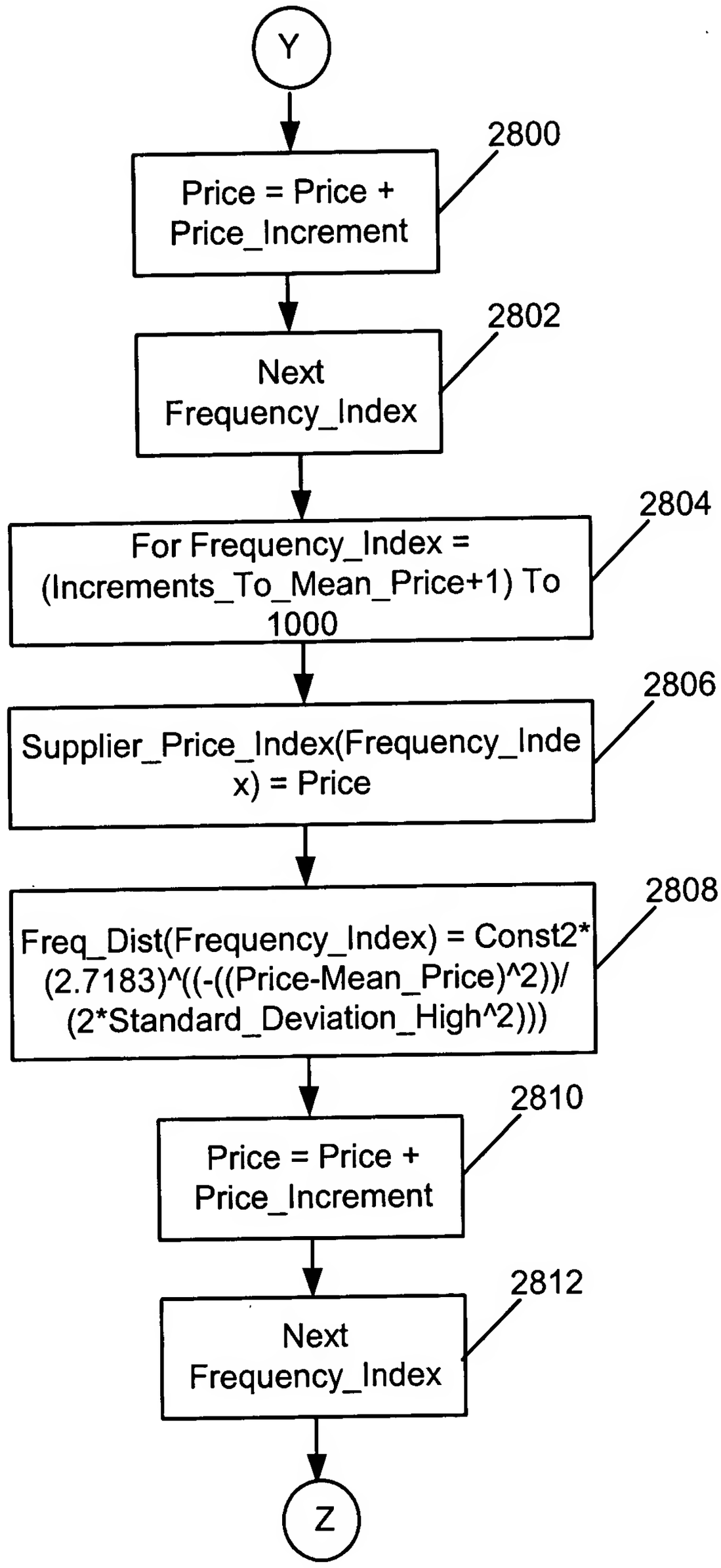

Y
Price = Price + Price_Increment
2800
Next Frequency_Index
2802
For Frequency_Index = (Increments_To_Mean_Price+1) To 1000
2804
Supplier_Price_Index(Frequency_Index) = Price
2806
Freq_Dist(Frequency_Index) = Const2* (2.7183)^((-((Price-Mean_Price)^2))/ (2*Standard_Deviation_High^2)))
2808
Price = Price + Price_Increment
2810
Next Frequency_Index
2812
Z

Fig 29

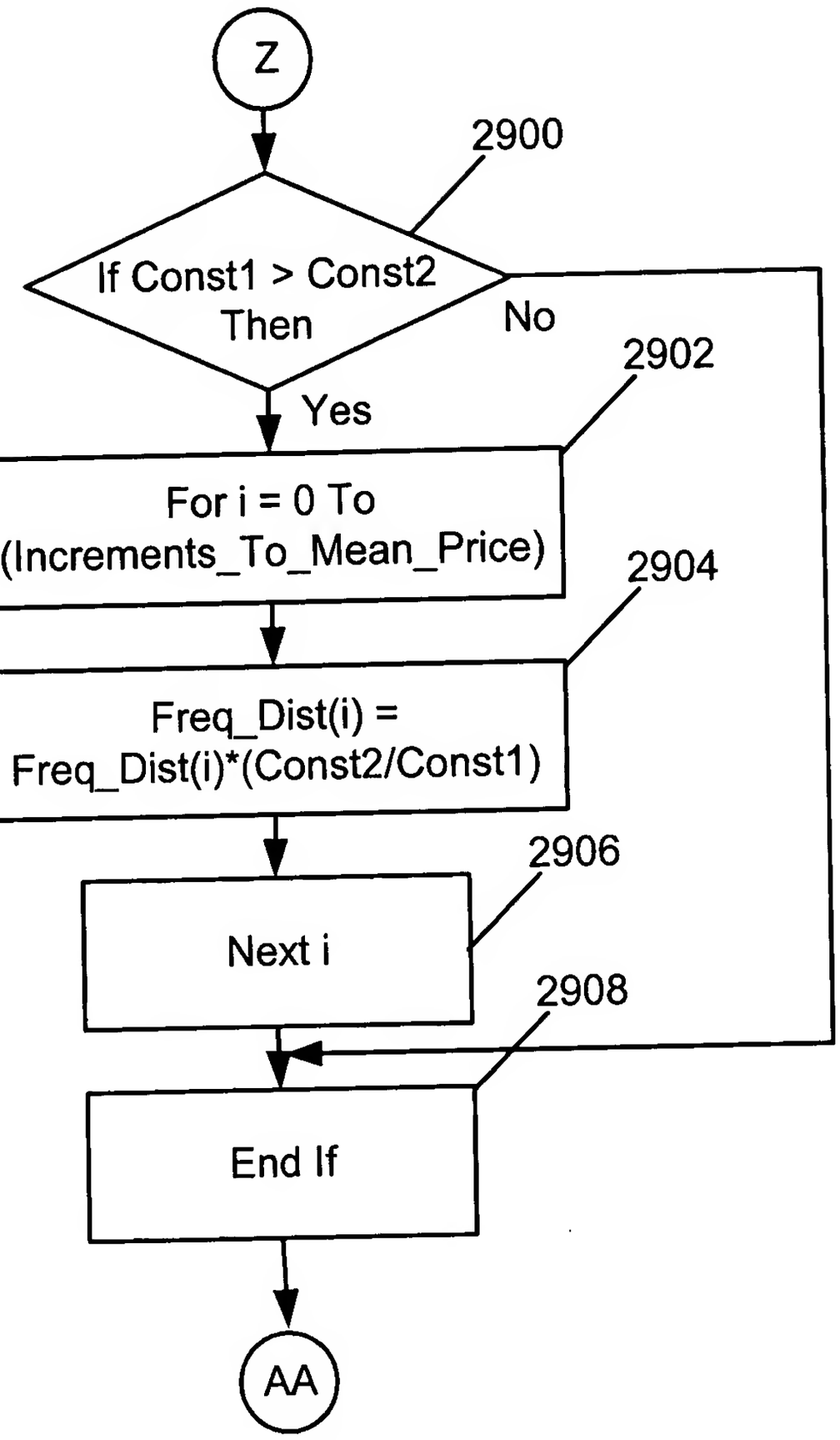

Z
2900
If Const1 > Const2
Then
No
Yes
2902
For i = 0 To
(Increments_To_Mean_Price)
2904
Freq_Dist(i) =
Freq_Dist(i)*(Const2/Const1)
2906
Next i
2908
End If
AA

Fig 30

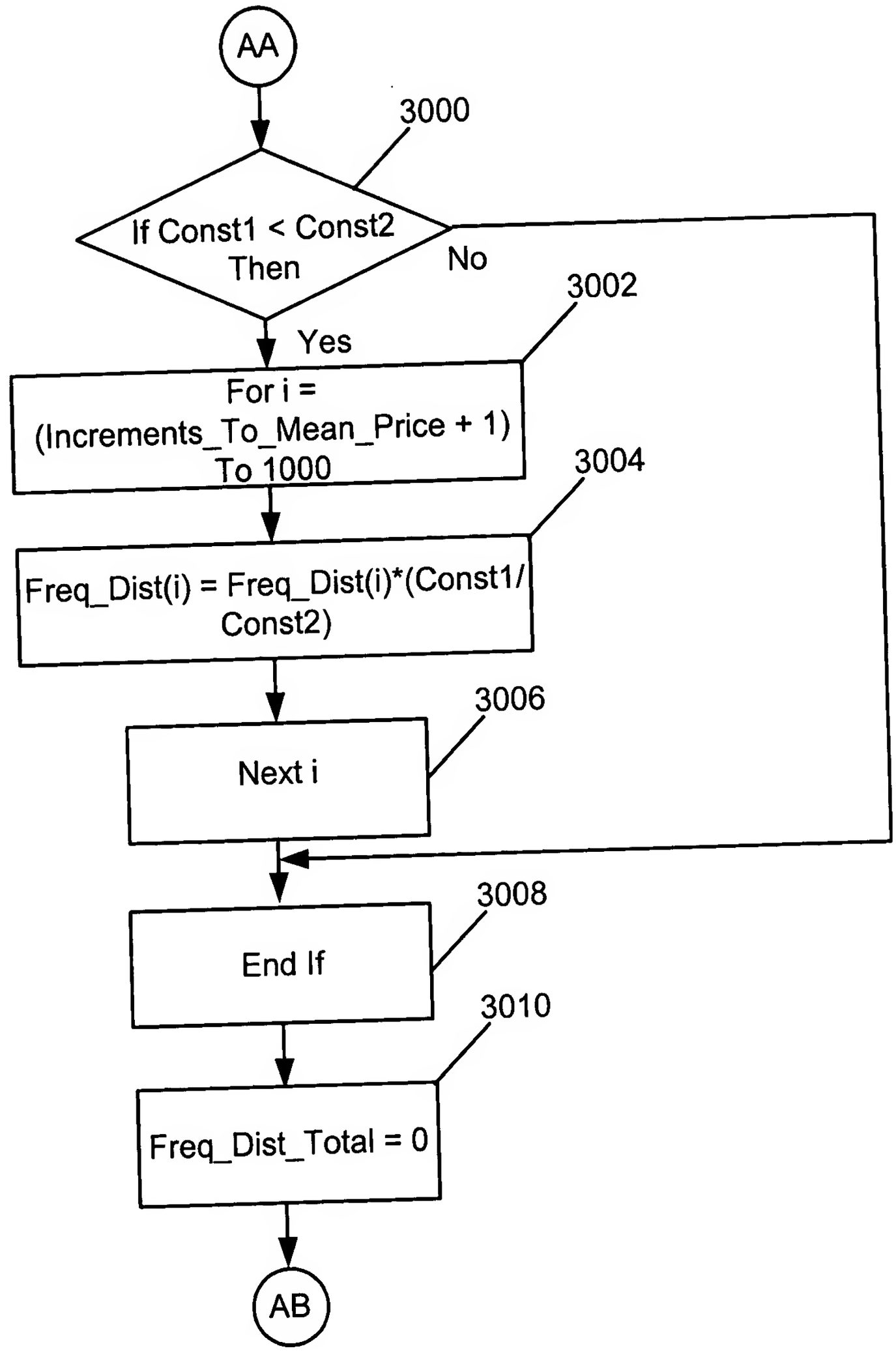

AA
3000
If Const1 < Const2
Then
No
Yes
3002
For i =
(Increments_To_Mean_Price + 1)
To 1000
3004
Freq_Dist(i) = Freq_Dist(i)*(Const1/
Const2)
3006
Next i
3008
End If
3010
Freq_Dist_Total = 0
AB

Fig 31

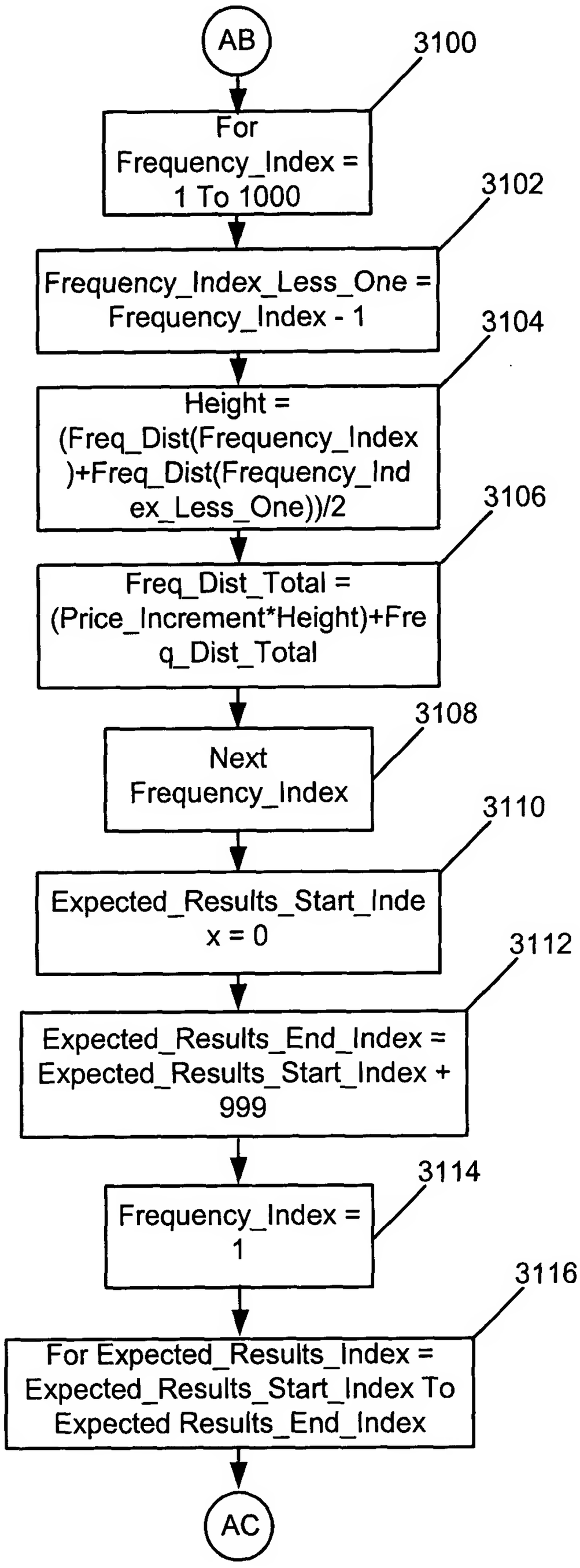

AB
3100
For Frequency_Index = 1 To 1000
3102
Frequency_Index_Less_One = Frequency_Index - 1
3104
Height = (Freq_Dist(Frequency_Index)+Freq_Dist(Frequency_Index_Less_One))/2
3106
Freq_Dist_Total = (Price_Increment*Height)+Freq_Dist_Total
3108
Next Frequency_Index
3110
Expected_Results_Start_Index = 0
3112
Expected_Results_End_Index = Expected_Results_Start_Index + 999
3114
Frequency_Index = 1
3116
For Expected_Results_Index = Expected_Results_Start_Index To Expected Results_End_Index
AC

Fig 32

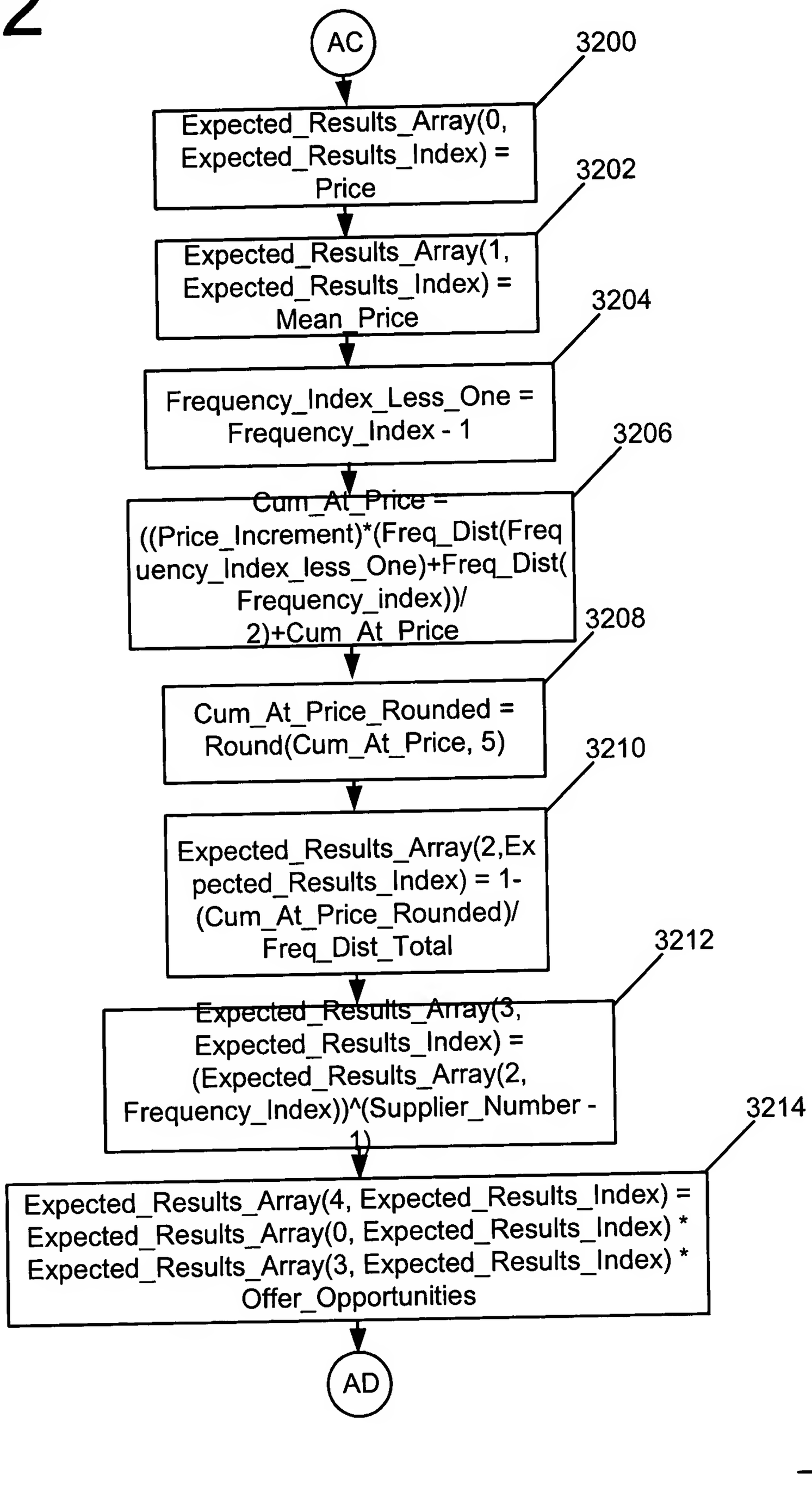

AC
3200
Expected_Results_Array(0, Expected_Results_Index) = Price
3202
Expected_Results_Array(1, Expected_Results_Index) = Mean_Price
3204
Frequency_Index_Less_One = Frequency_Index - 1
3206
Cum_At_Price = ((Price_Increment)*(Freq_Dist(Frequency_Index_less_One)+Freq_Dist(Frequency_index))/2)+Cum_At_Price
3208
Cum_At_Price_Rounded = Round(Cum_At_Price, 5)
3210
Expected_Results_Array(2,Expected_Results_Index) = 1-(Cum_At_Price_Rounded)/Freq_Dist_Total
3212
Expected_Results_Array(3, Expected_Results_Index) = (Expected_Results_Array(2, Frequency_Index))^(Supplier_Number - 1)
3214
Expected_Results_Array(4, Expected_Results_Index) = Expected_Results_Array(0, Expected_Results_Index) * Expected_Results_Array(3, Expected_Results_Index) * Offer_Opportunities
AD

Fig 33

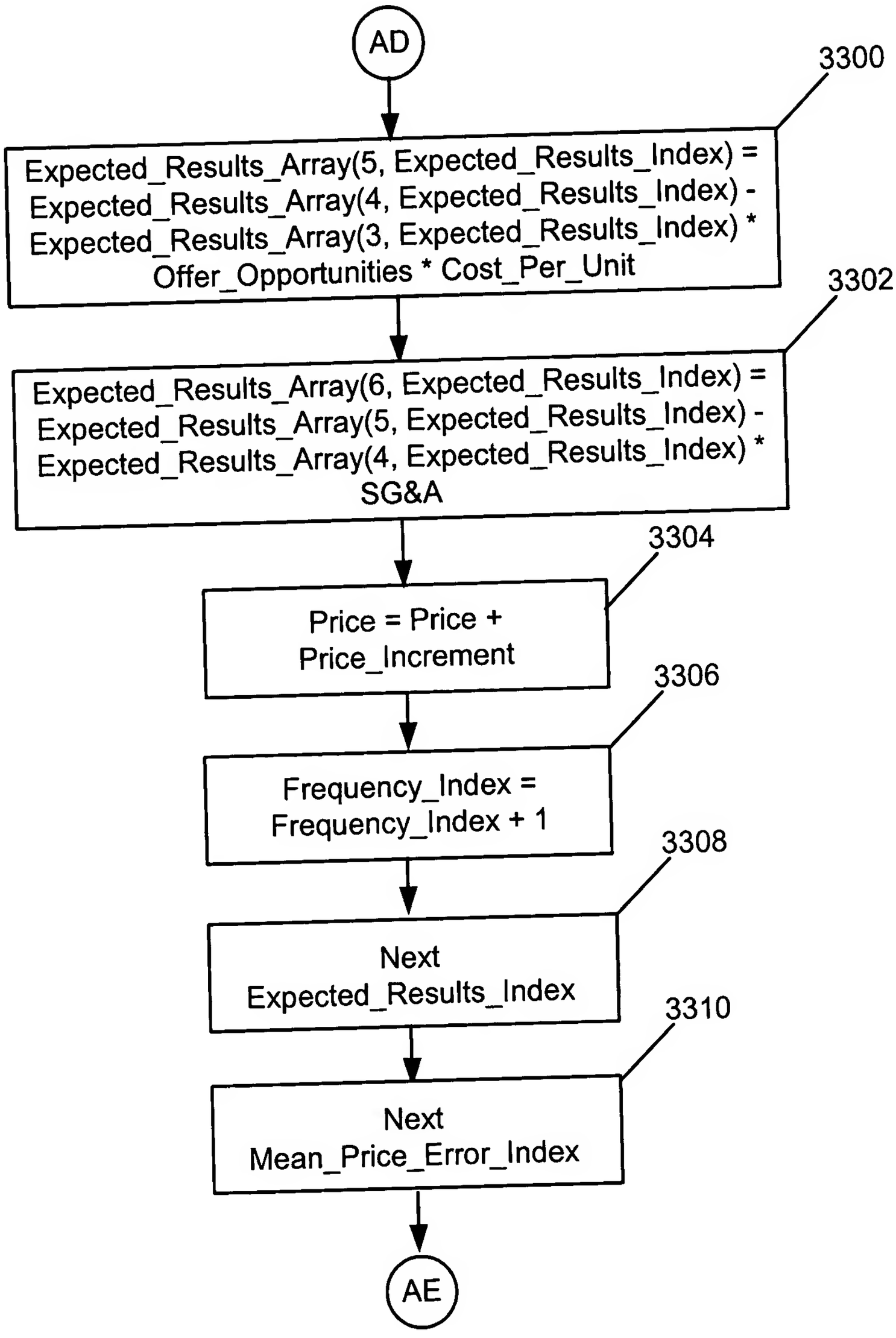

AD
3300
Expected_Results_Array(5, Expected_Results_Index) = Expected_Results_Array(4, Expected_Results_Index) - Expected_Results_Array(3, Expected_Results_Index) * Offer_Opportunities * Cost_Per_Unit
3302
Expected_Results_Array(6, Expected_Results_Index) = Expected_Results_Array(5, Expected_Results_Index) - Expected_Results_Array(4, Expected_Results_Index) * SG&A
3304
Price = Price + Price_Increment
3306
Frequency_Index = Frequency_Index + 1
3308
Next Expected_Results_Index
3310
Next Mean_Price_Error_Index
AE

Fig 34

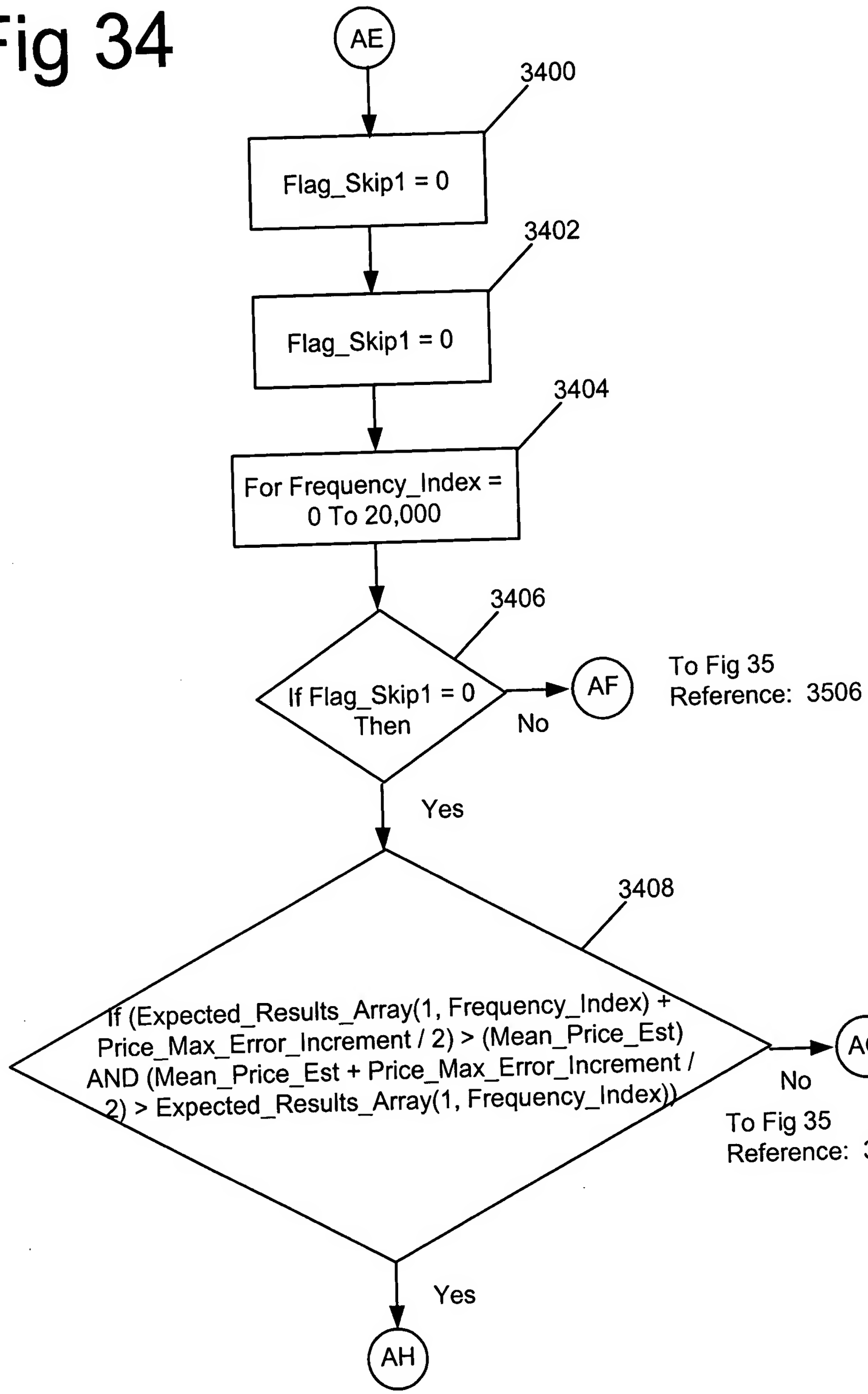

AE
3400
Flag_Skip1 = 0
3402
Flag_Skip1 = 0
3404
For Frequency_Index = 0 To 20,000
3406
If Flag_Skip1 = 0 Then
No
AF
To Fig 35 Reference: 3506
Yes
3408
If (Expected_Results_Array(1, Frequency_Index) + Price_Max_Error_Increment / 2) > (Mean_Price_Est) AND (Mean_Price_Est + Price_Max_Error_Increment / 2) > Expected_Results_Array(1, Frequency_Index))
No
AG
To Fig 35 Reference: 3504
Yes
AH

Fig 35

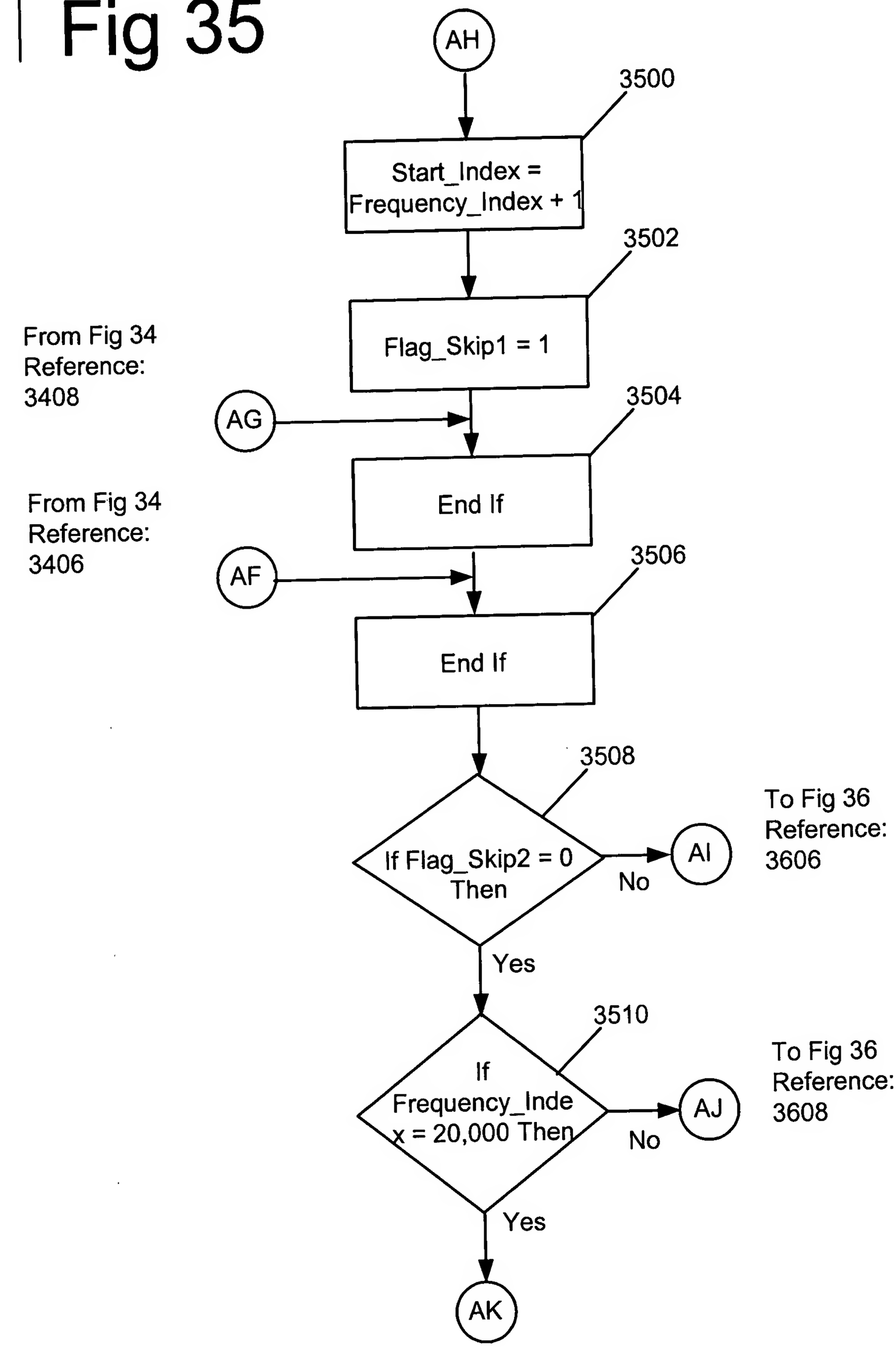

From Fig 34
Reference:
3408
AH
3500
Start_Index =
Frequency_Index + 1
3502
Flag_Skip1 = 1
AG
3504
End If
From Fig 34
Reference:
3406
AF
3506
End If
3508
If Flag_Skip2 = 0
Then
No
AI
To Fig 36
Reference:
3606
Yes
3510
If
Frequency_Inde
x = 20,000 Then
No
AJ
To Fig 36
Reference:
3608
Yes
AK

Fig 36

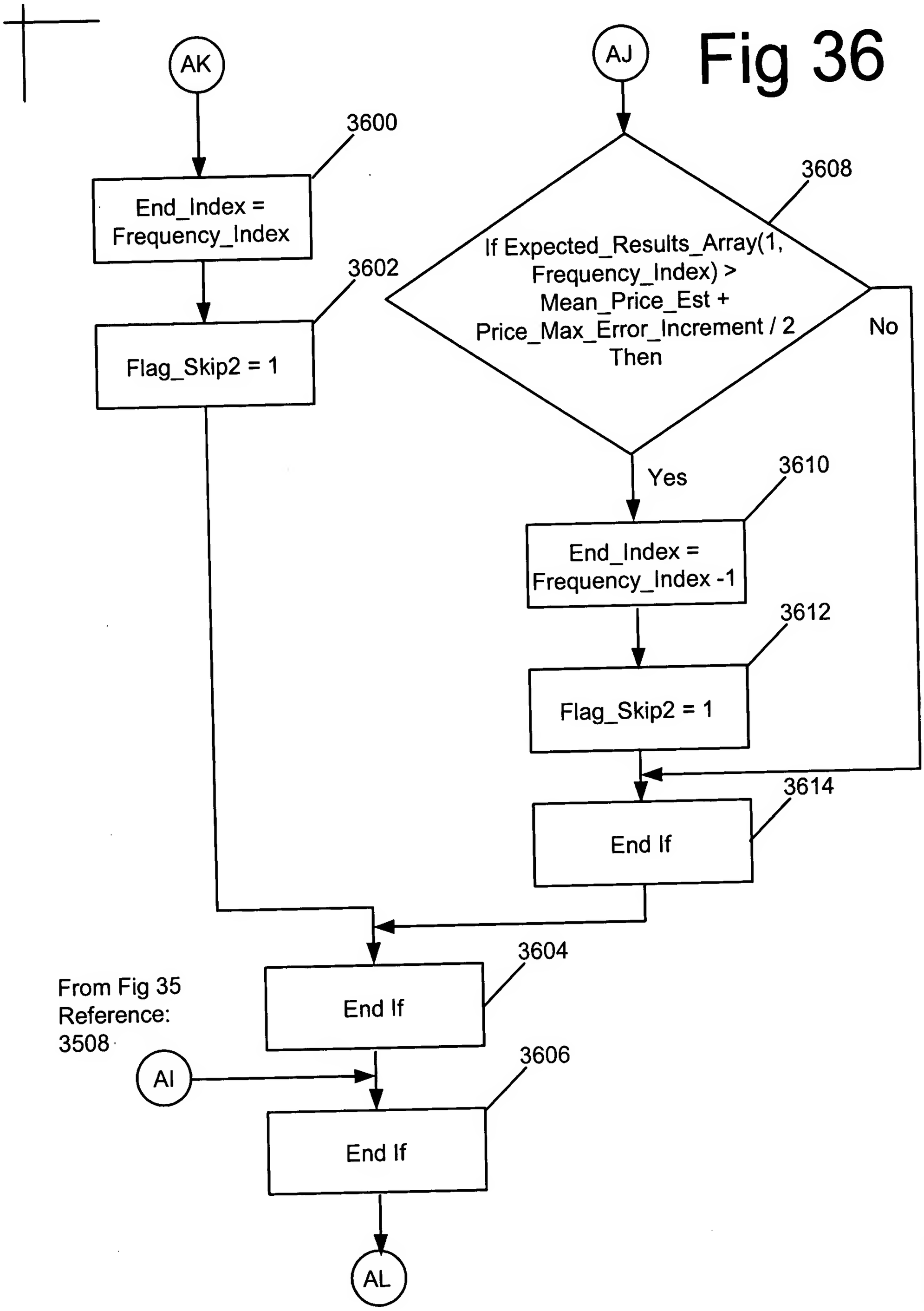

AK
3600
End_Index =
Frequency_Index
3602
Flag_Skip2 = 1
AJ
3608
If Expected_Results_Array(1,
Frequency_Index) >
Mean_Price_Est +
Price_Max_Error_Increment / 2
Then
No
Yes
3610
End_Index =
Frequency_Index -1
3612
Flag_Skip2 = 1
3614
End If
3604
End If
From Fig 35
Reference:
3508
AI
3606
End If
AL

Fig 37

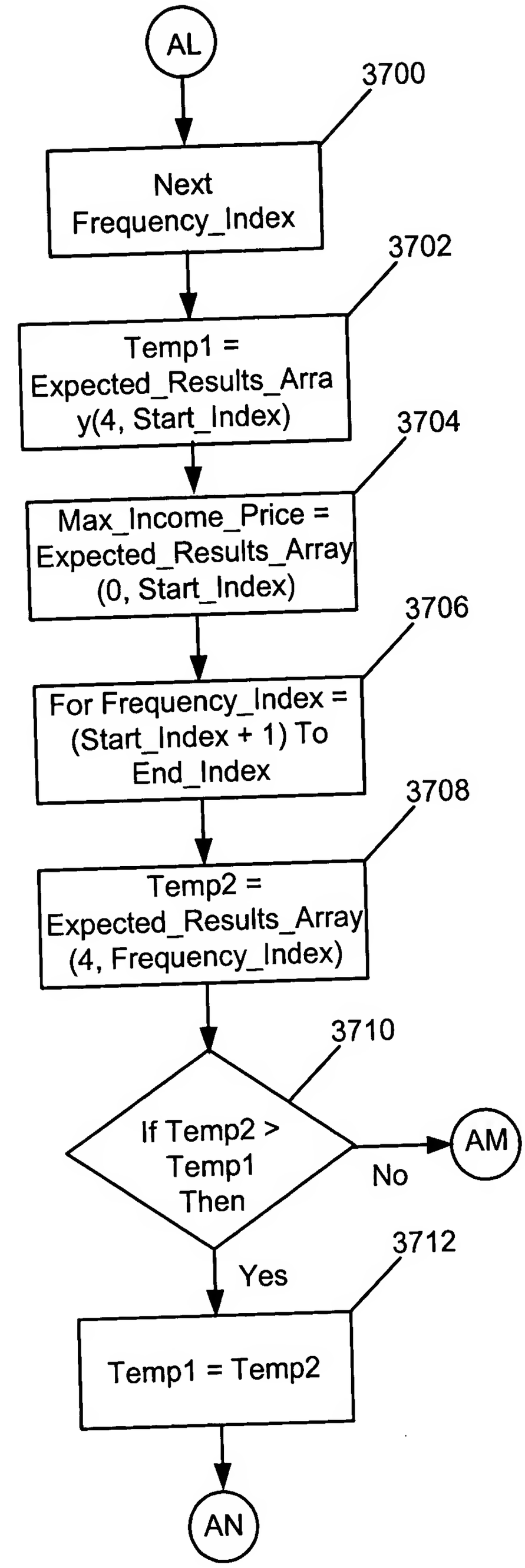


Fig 38
Reference: 3804

Fig 38

From Fig 37
Reference: 3710

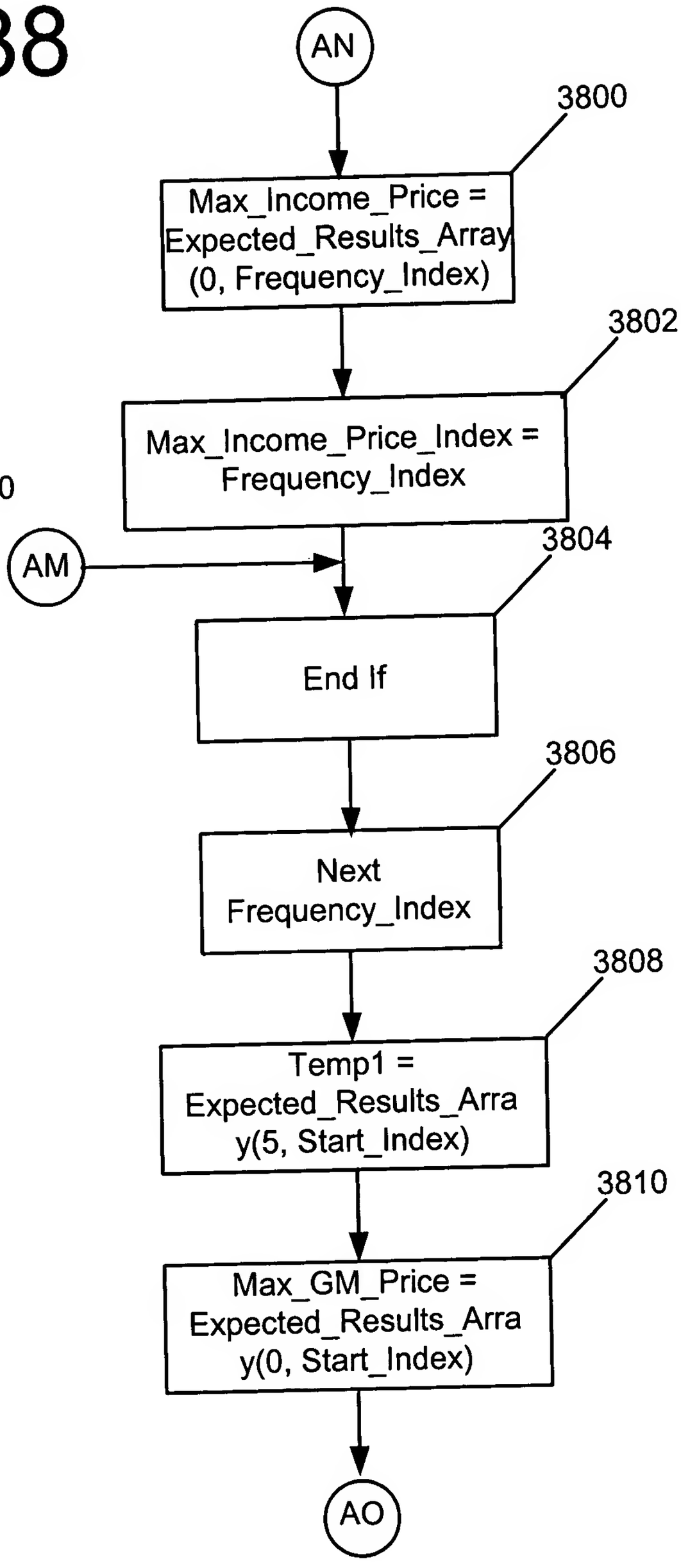


Fig 39

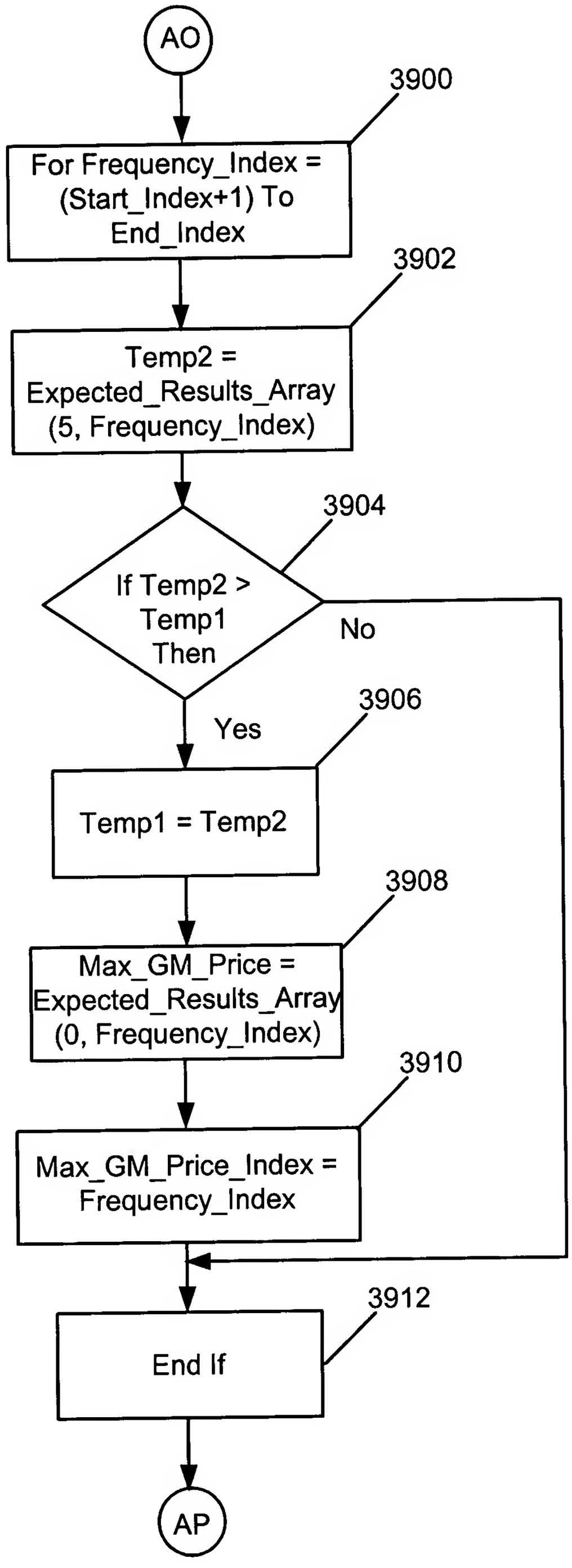

AO
3900
For Frequency_Index = (Start_Index+1) To End_Index
3902
Temp2 = Expected_Results_Array (5, Frequency_Index)
3904
If Temp2 > Temp1 Then
No
Yes
3906
Temp1 = Temp2
3908
Max_GM_Price = Expected_Results_Array (0, Frequency_Index)
3910
Max_GM_Price_Index = Frequency_Index
3912
End If
AP

Fig 40

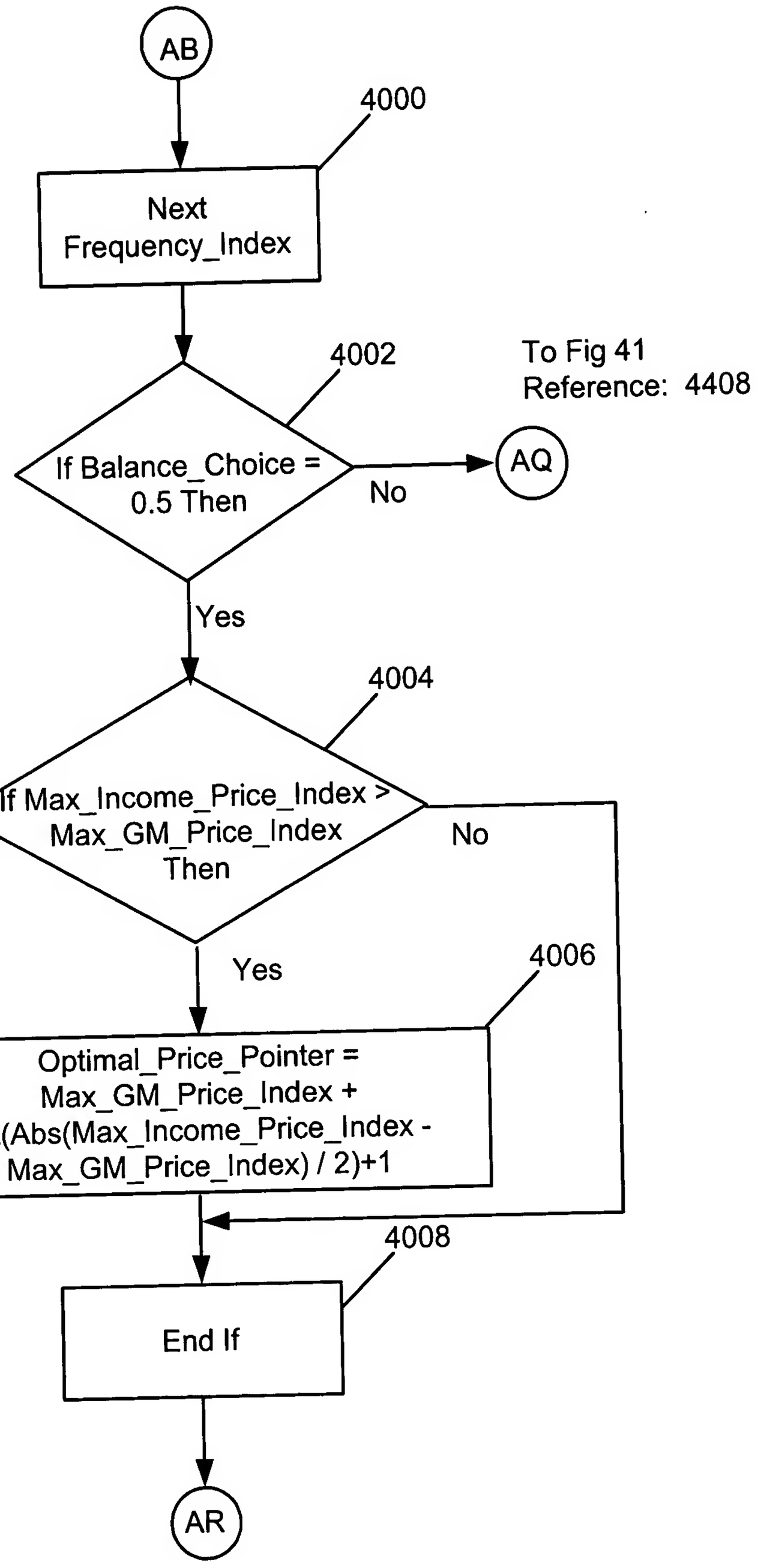

AB
4000
Next
Frequency_Index
4002
To Fig 41
Reference: 4408
If Balance_Choice =
0.5 Then
No
AQ
Yes
4004
If Max_Income_Price_Index >
Max_GM_Price_Index
Then
No
Yes
4006
Optimal_Price_Pointer =
Max_GM_Price_Index +
Int(Abs(Max_Income_Price_Index -
Max_GM_Price_Index) / 2)+1
4008
End If
AR

Fig 41

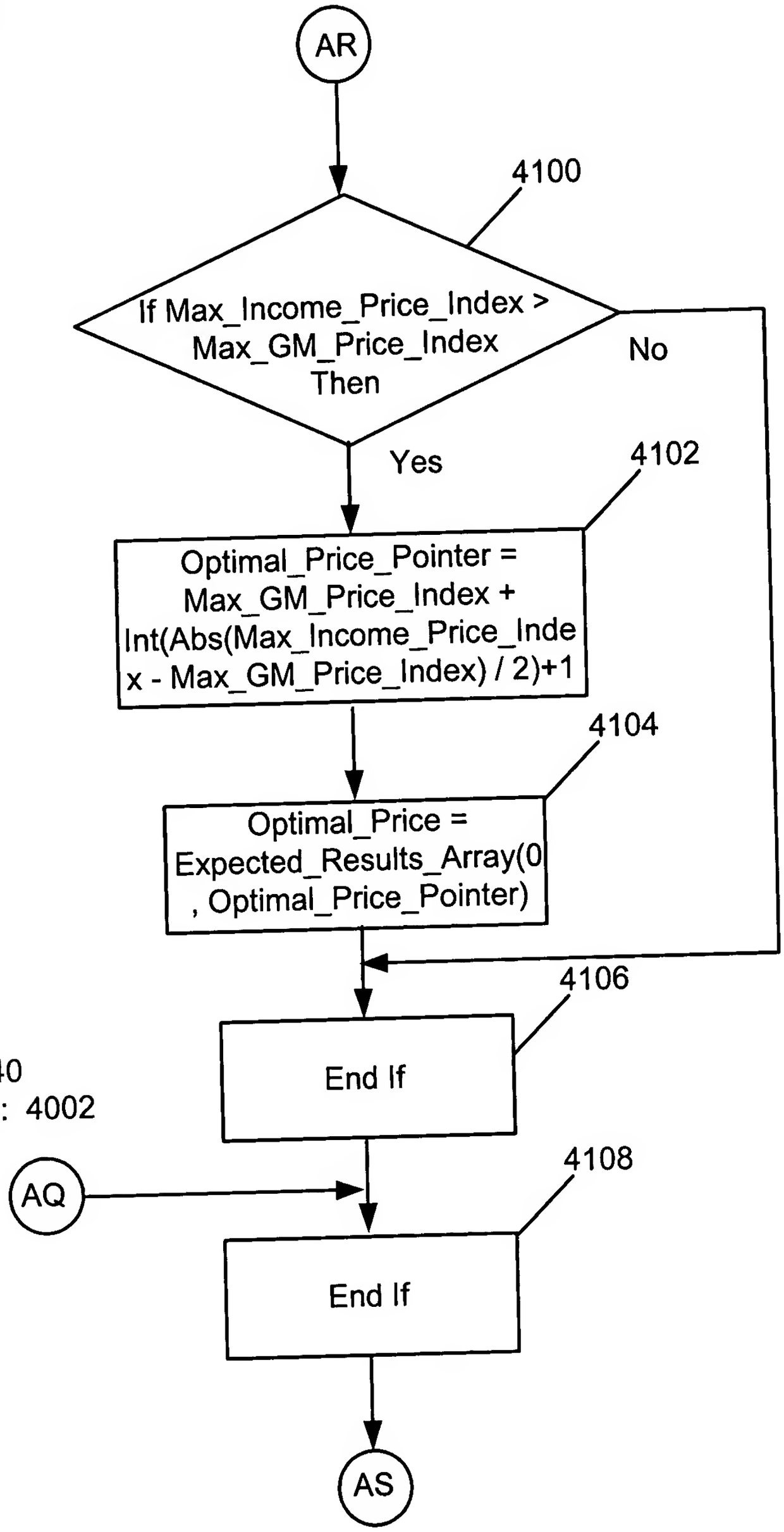

AR
4100
If Max_Income_Price_Index > Max_GM_Price_Index Then
No
Yes
4102
Optimal_Price_Pointer = Max_GM_Price_Index + Int(Abs(Max_Income_Price_Index - Max_GM_Price_Index) / 2)+1
4104
Optimal_Price = Expected_Results_Array(0, Optimal_Price_Pointer)
4106
End If
From Fig 40
Reference: 4002
AQ
4108
End If
AS

Fig 42

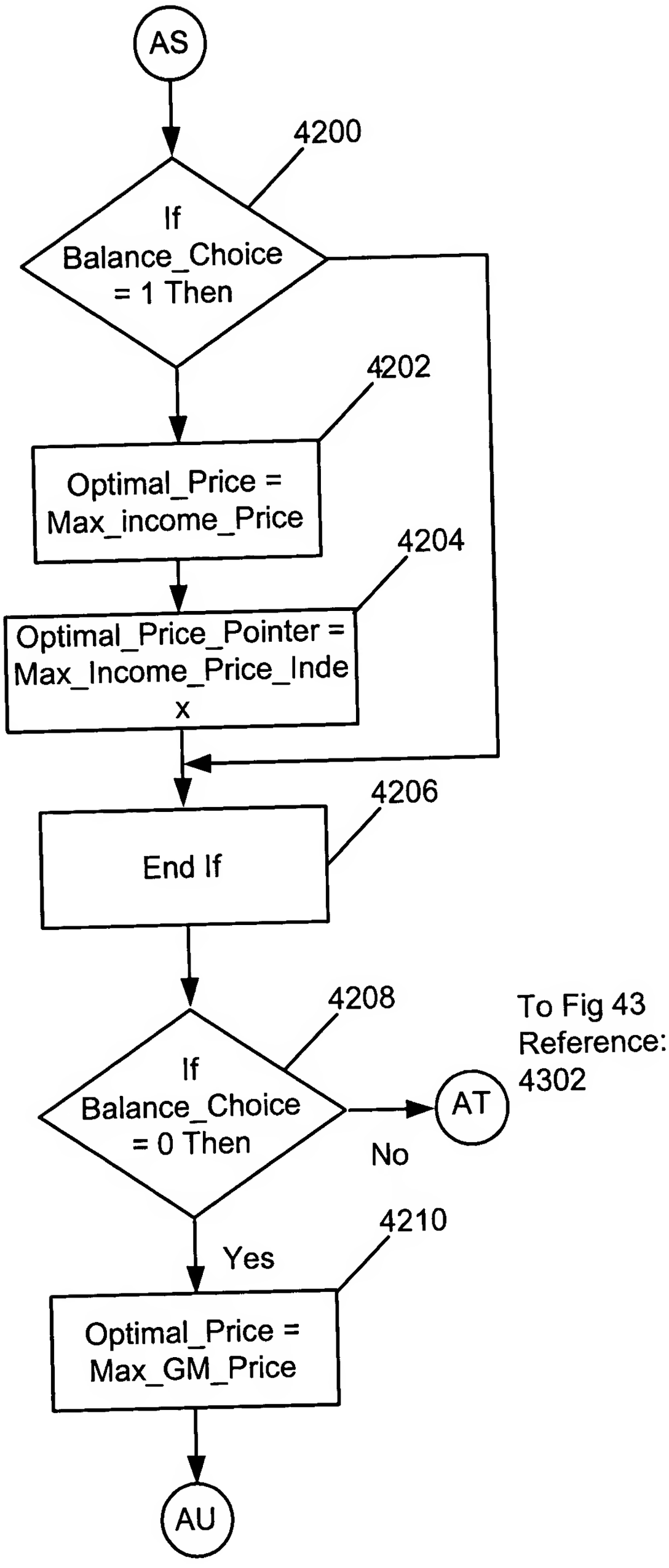

AS
4200
If
Balance_Choice
= 1 Then
4202
Optimal_Price =
Max_income_Price
4204
Optimal_Price_Pointer =
Max_Income_Price_Inde
x
4206
End If
4208
If
Balance_Choice
= 0 Then
No
AT
To Fig 43
Reference:
4302
Yes
4210
Optimal_Price =
Max_GM_Price
AU

Fig 43

From Fig 42
Reference: 4208

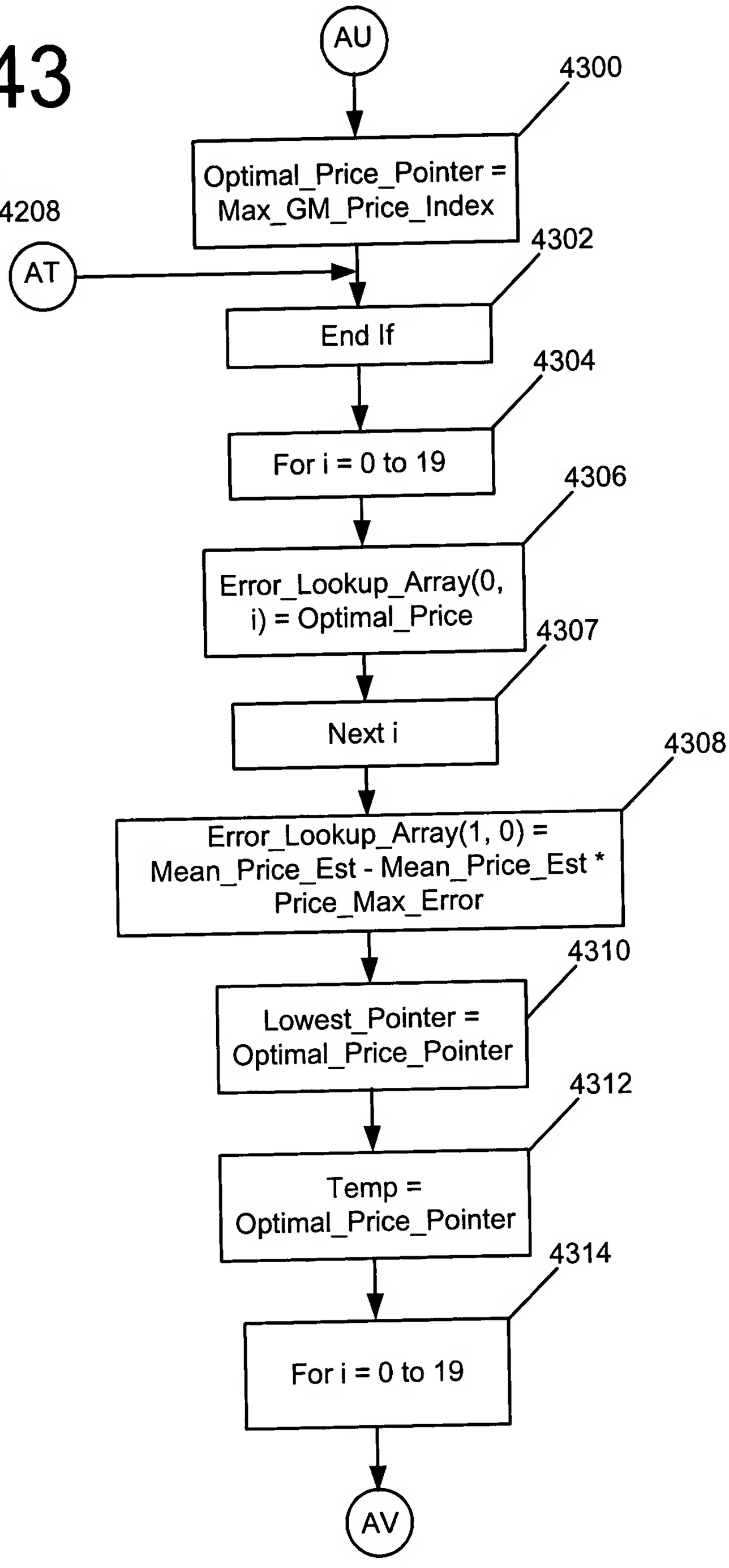


Fig 44

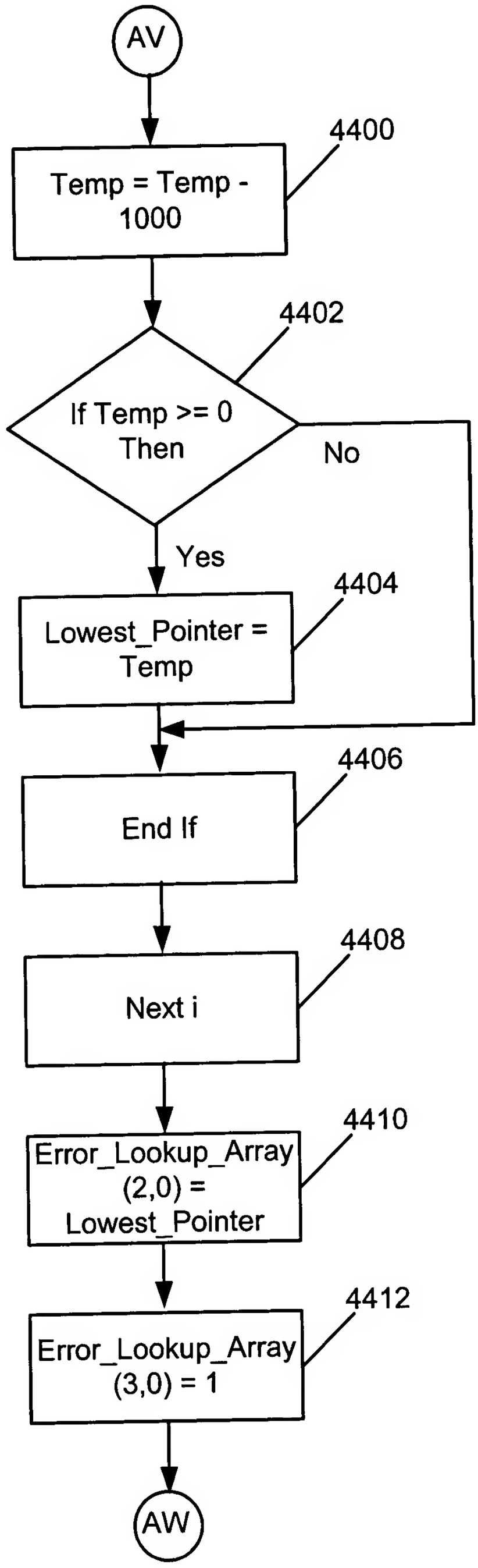

AV
Temp = Temp - 1000
4400
If Temp >= 0 Then
4402
No
Yes
Lowest_Pointer = Temp
4404
End If
4406
Next i
4408
Error_Lookup_Array (2,0) = Lowest_Pointer
4410
Error_Lookup_Array (3,0) = 1
4412
AW

Fig 45

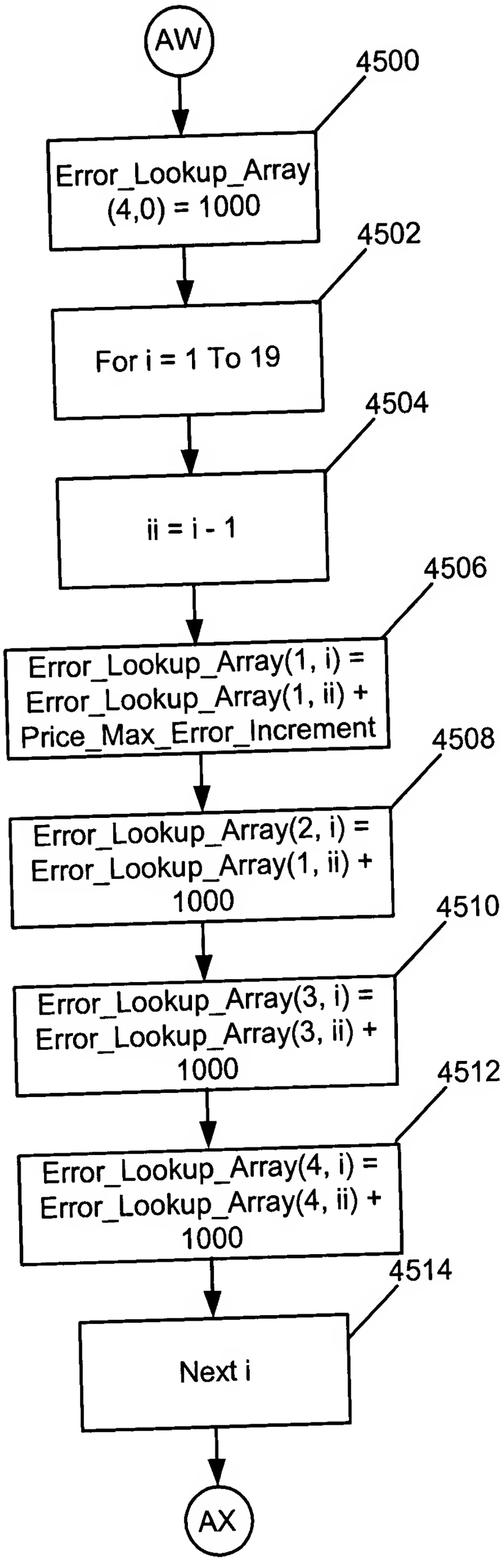

AW
4500
Error_Lookup_Array
(4,0) = 1000
4502
For i = 1 To 19
4504
ii = i - 1
4506
Error_Lookup_Array(1, i) =
Error_Lookup_Array(1, ii) +
Price_Max_Error_Increment
4508
Error_Lookup_Array(2, i) =
Error_Lookup_Array(1, ii) +
1000
4510
Error_Lookup_Array(3, i) =
Error_Lookup_Array(3, ii) +
1000
4512
Error_Lookup_Array(4, i) =
Error_Lookup_Array(4, ii) +
1000
4514
Next i
AX

Fig 46

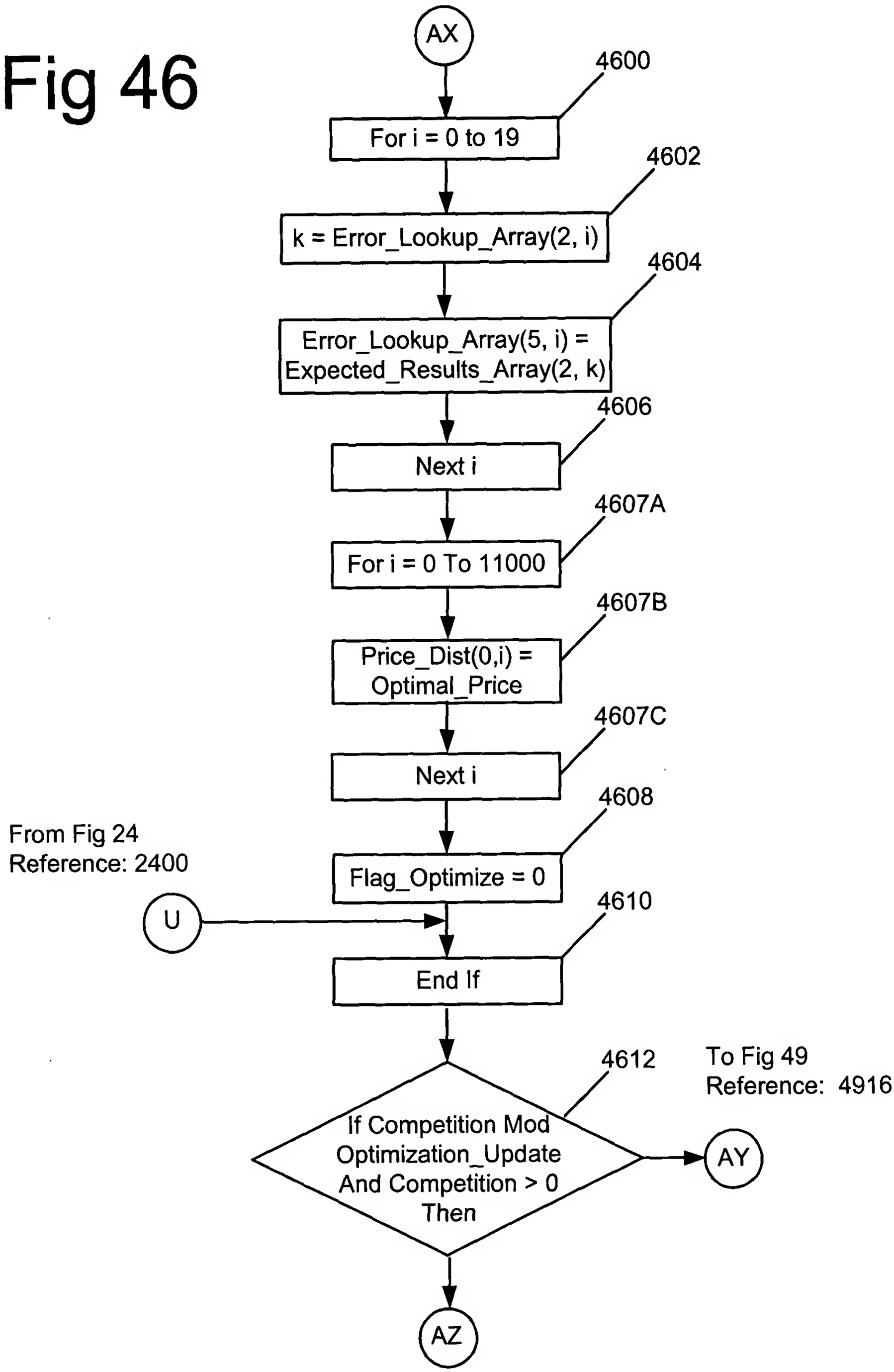

AX
4600
For i = 0 to 19
4602
k = Error_Lookup_Array(2, i)
4604
Error_Lookup_Array(5, i) =
Expected_Results_Array(2, k)
4606
Next i
4607A
For i = 0 To 11000
4607B
Price_Dist(0,i) =
Optimal_Price
4607C
Next i
4608
Flag_Optimize = 0
From Fig 24
Reference: 2400
U
4610
End If
4612
If Competition Mod
Optimization_Update
And Competition > 0
Then
AY
To Fig 49
Reference: 4916
AZ

Fig 47

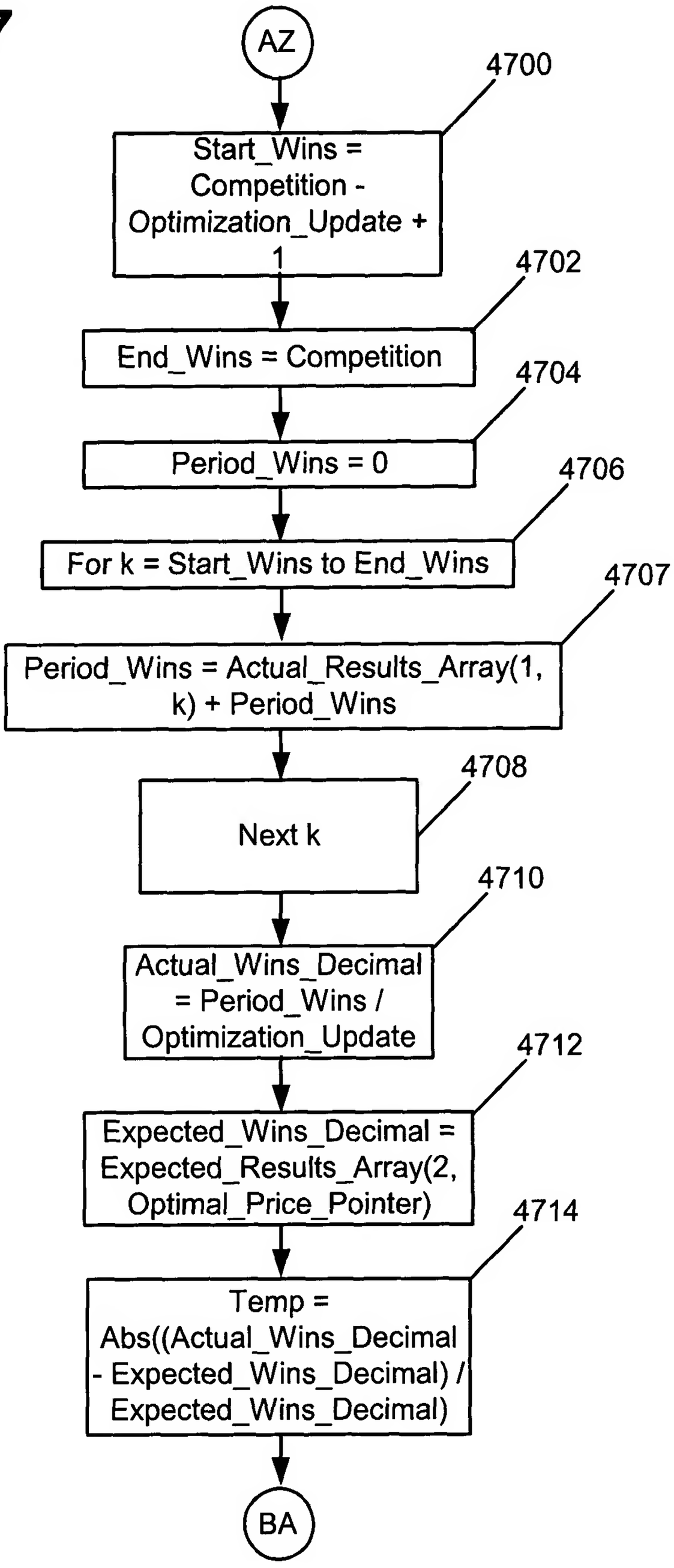

AZ
4700
Start_Wins = Competition - Optimization_Update + 1
4702
End_Wins = Competition
4704
Period_Wins = 0
4706
For k = Start_Wins to End_Wins
4707
Period_Wins = Actual_Results_Array(1, k) + Period_Wins
4708
Next k
4710
Actual_Wins_Decimal = Period_Wins / Optimization_Update
4712
Expected_Wins_Decimal = Expected_Results_Array(2, Optimal_Price_Pointer)
4714
Temp = Abs((Actual_Wins_Decimal - Expected_Wins_Decimal) / Expected_Wins_Decimal)
BA

Fig 48

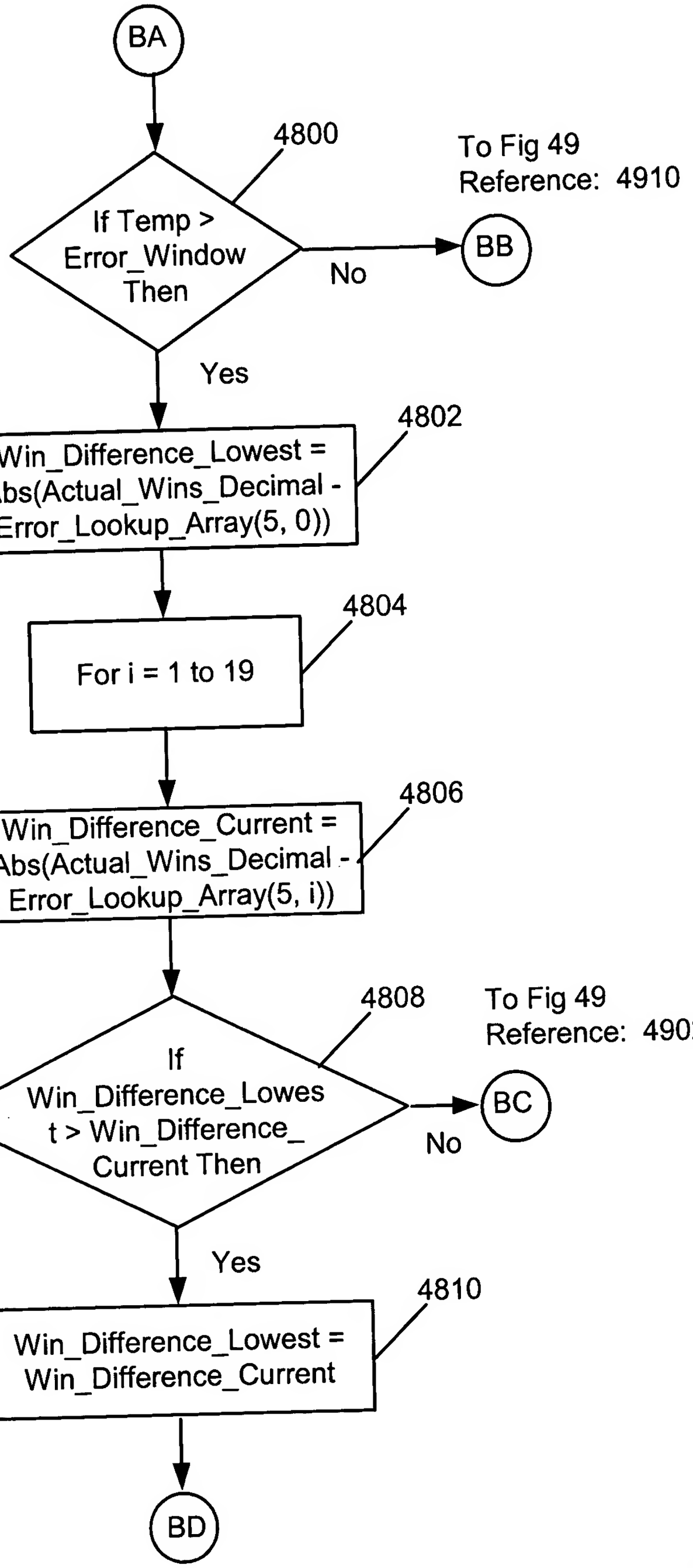

BA
4800
To Fig 49
Reference: 4910
If Temp > Error_Window Then
No
BB
Yes
4802
Win_Difference_Lowest = Abs(Actual_Wins_Decimal - Error_Lookup_Array(5, 0))
4804
For i = 1 to 19
4806
Win_Difference_Current = Abs(Actual_Wins_Decimal - Error_Lookup_Array(5, i))
4808
To Fig 49
Reference: 4902
If Win_Difference_Lowest > Win_Difference_Current Then
No
BC
Yes
4810
Win_Difference_Lowest = Win_Difference_Current
BD

Fig 49

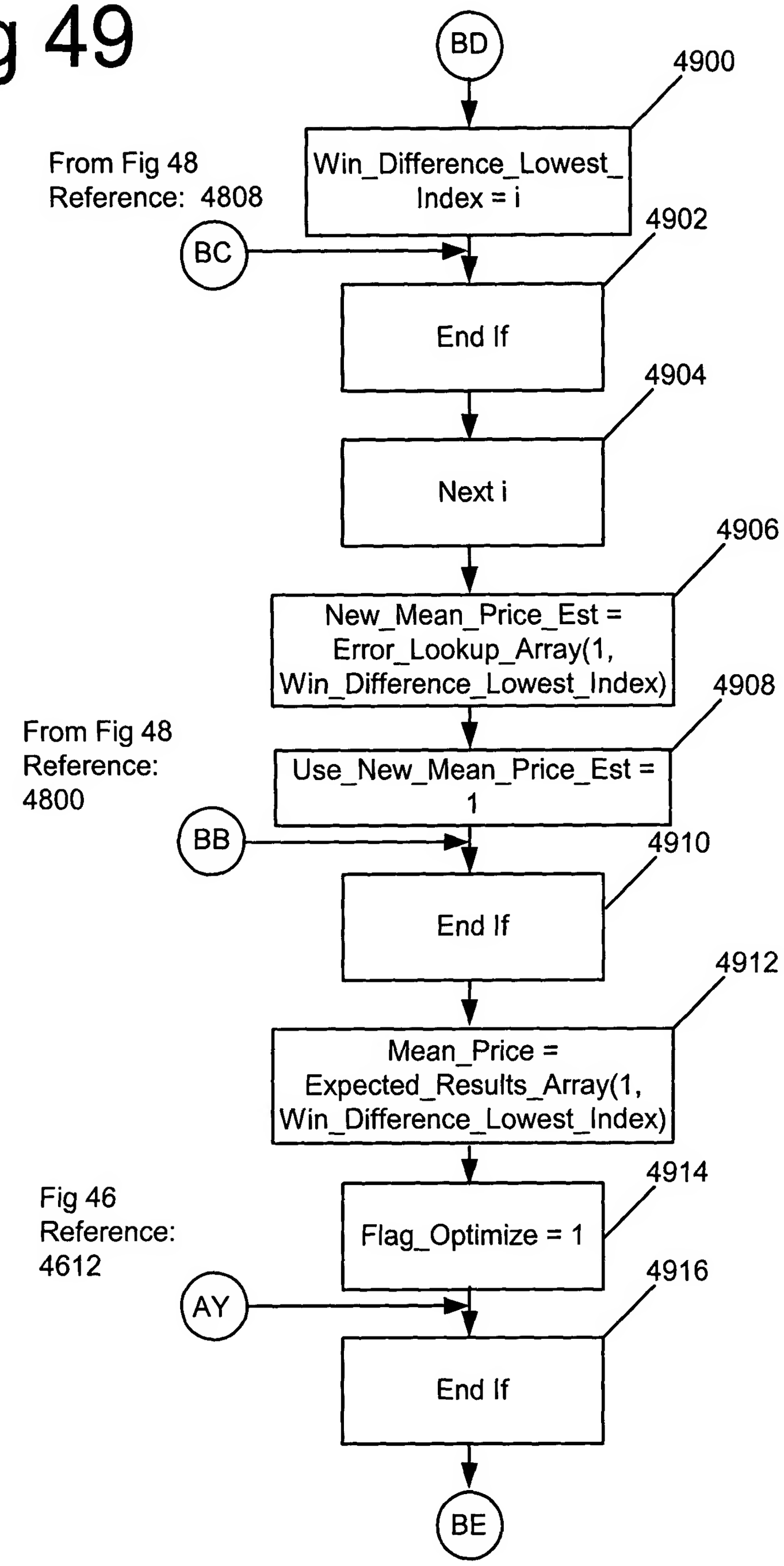

From Fig 48
Reference: 4808
BD
4900
Win_Difference_Lowest_
Index = i
BC
4902
End If
4904
Next i
4906
New_Mean_Price_Est =
Error_Lookup_Array(1,
Win_Difference_Lowest_Index)
4908
From Fig 48
Reference:
4800
Use_New_Mean_Price_Est =
1
BB
4910
End If
4912
Mean_Price =
Expected_Results_Array(1,
Win_Difference_Lowest_Index)
4914
Fig 46
Reference:
4612
Flag_Optimize = 1
4916
AY
End If
BE

Fig 50

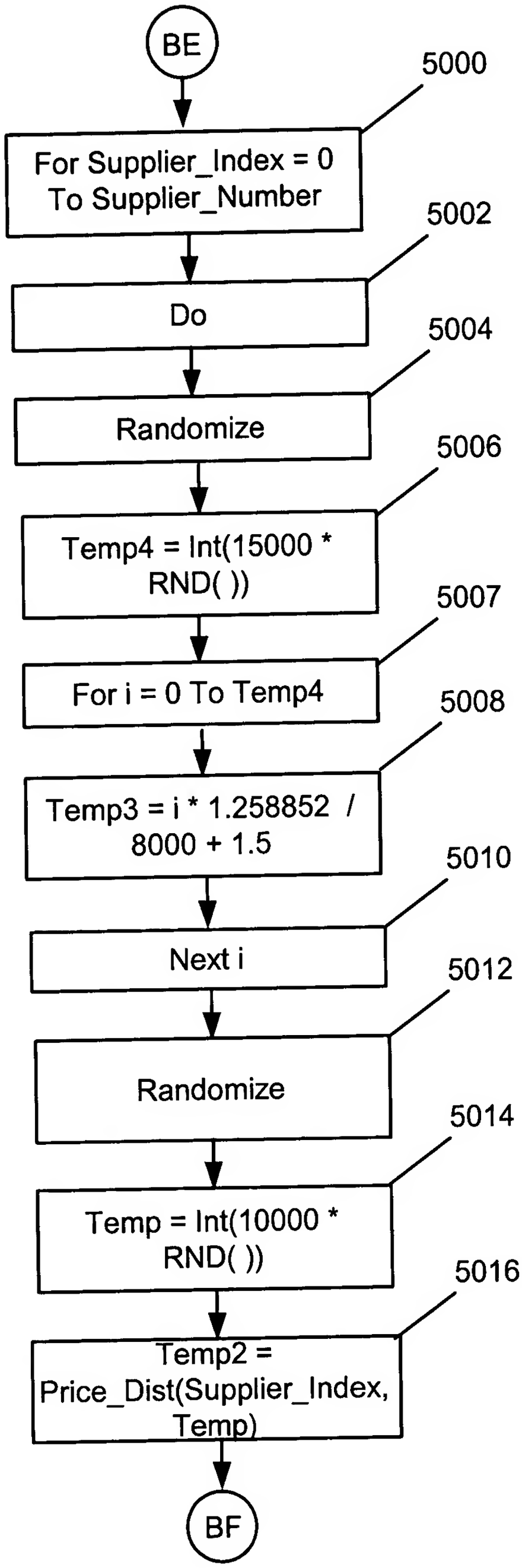

BE
5000
For Supplier_Index = 0 To Supplier_Number
5002
Do
5004
Randomize
5006
Temp4 = Int(15000 * RND( ))
5007
For i = 0 To Temp4
5008
Temp3 = i * 1.258852 / 8000 + 1.5
5010
Next i
5012
Randomize
5014
Temp = Int(10000 * RND( ))
5016
Temp2 = Price_Dist(Supplier_Index, Temp)
BF

Fig 51

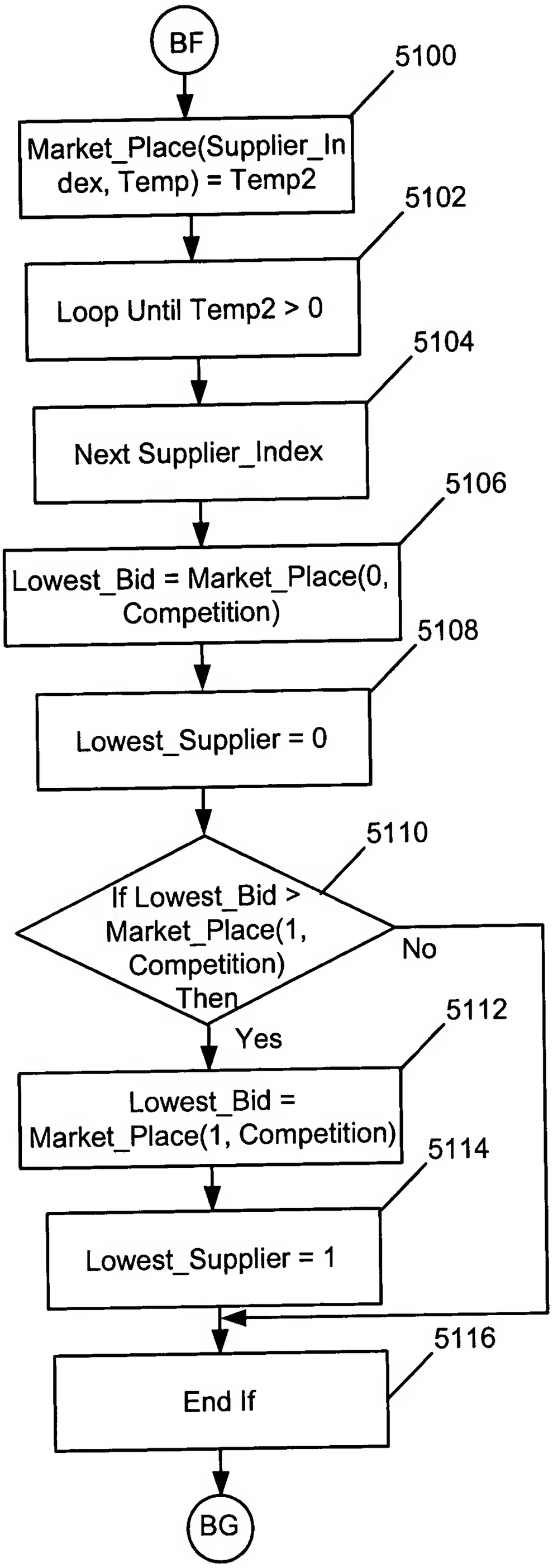

BF
5100
Market_Place(Supplier_Index, Temp) = Temp2
5102
Loop Until Temp2 > 0
5104
Next Supplier_Index
5106
Lowest_Bid = Market_Place(0, Competition)
5108
Lowest_Supplier = 0
5110
If Lowest_Bid > Market_Place(1, Competition) Then
No
Yes
5112
Lowest_Bid = Market_Place(1, Competition)
5114
Lowest_Supplier = 1
5116
End If
BG

Fig 52

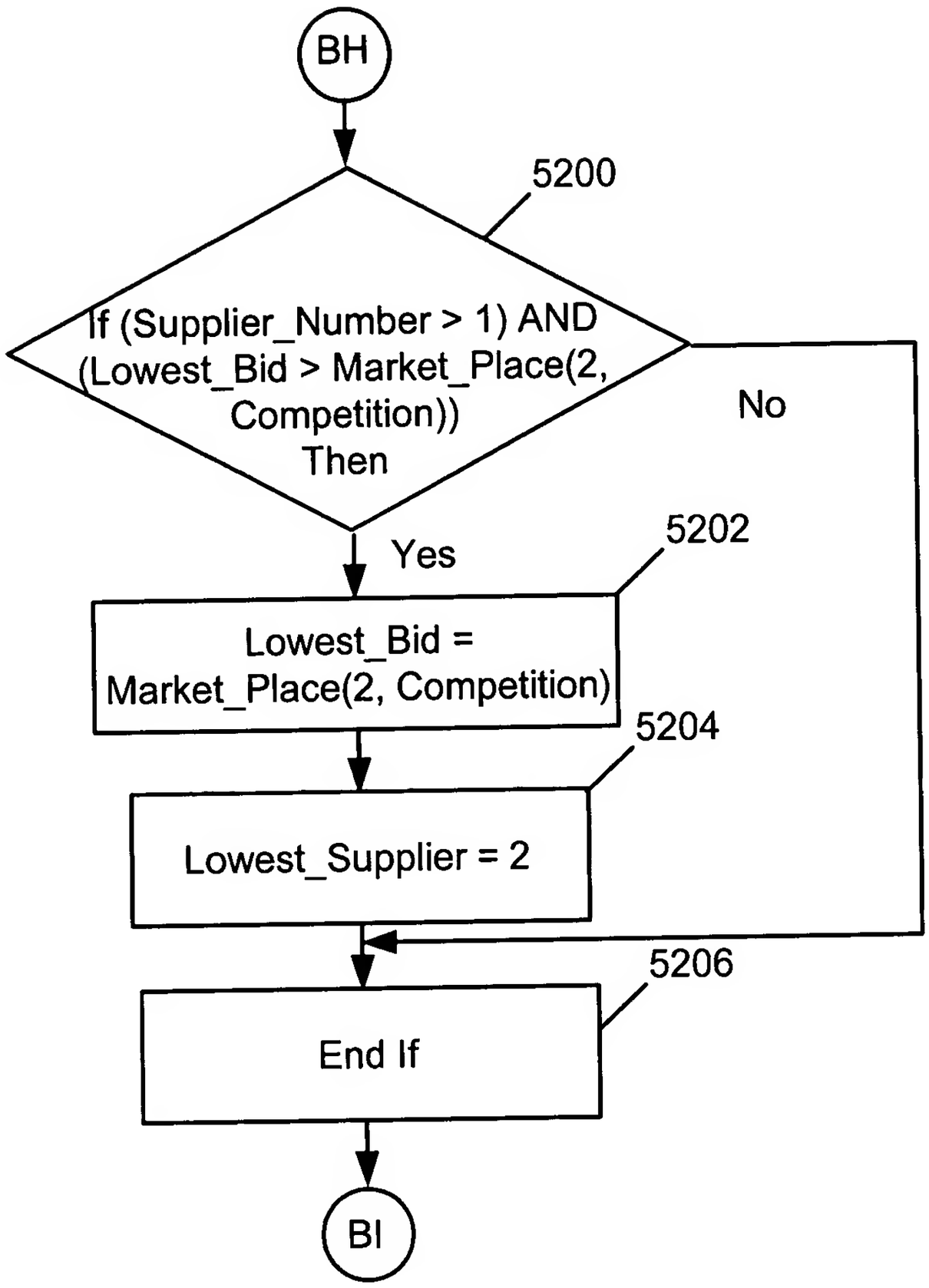

BH
5200
If (Supplier_Number > 1) AND (Lowest_Bid > Market_Place(2, Competition)) Then
No
Yes
5202
Lowest_Bid = Market_Place(2, Competition)
5204
Lowest_Supplier = 2
5206
End If
BI

Fig 53

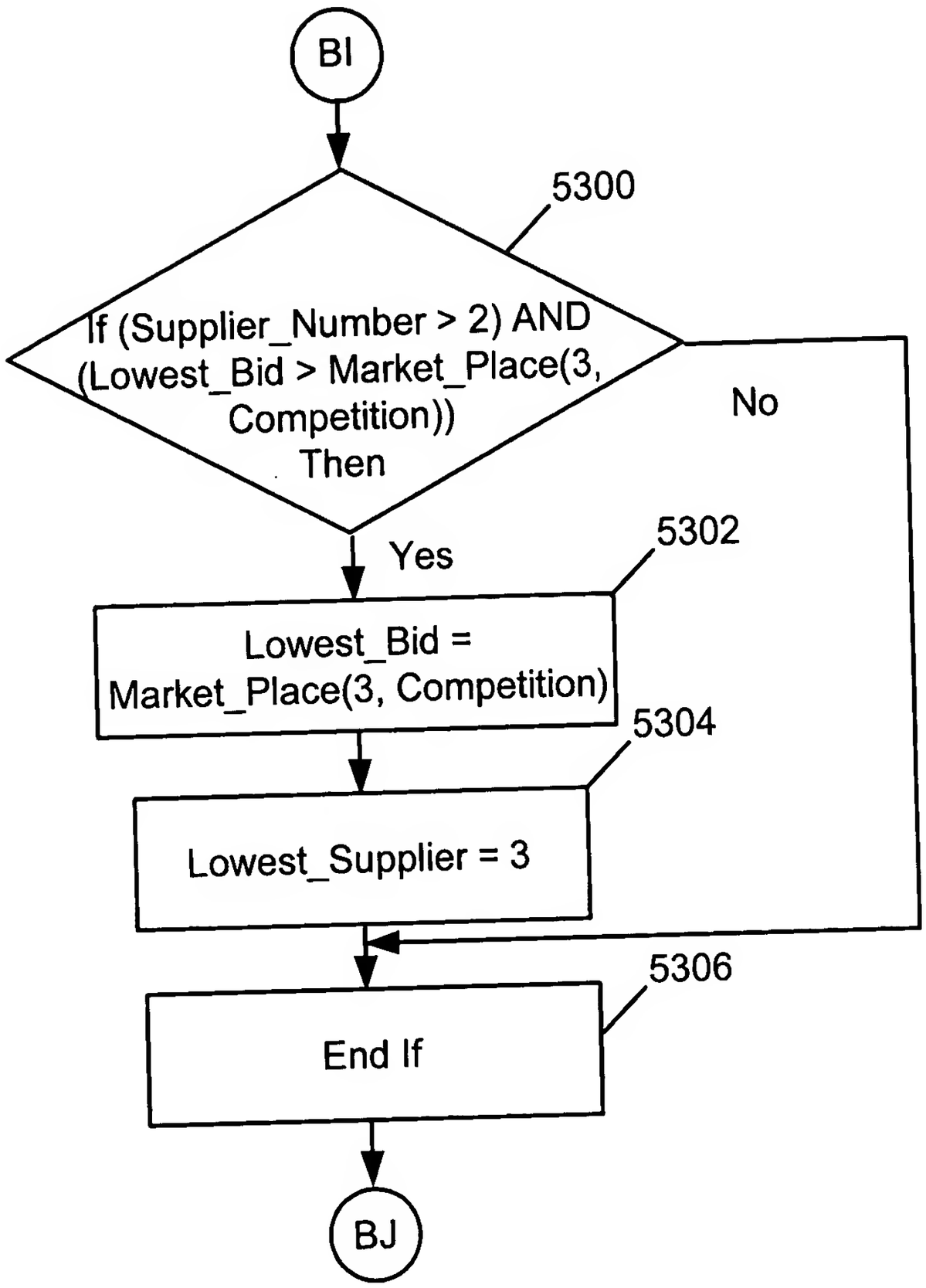

BI
5300
If (Supplier_Number > 2) AND (Lowest_Bid > Market_Place(3, Competition)) Then
No
Yes
5302
Lowest_Bid = Market_Place(3, Competition)
5304
Lowest_Supplier = 3
5306
End If
BJ

Fig 54

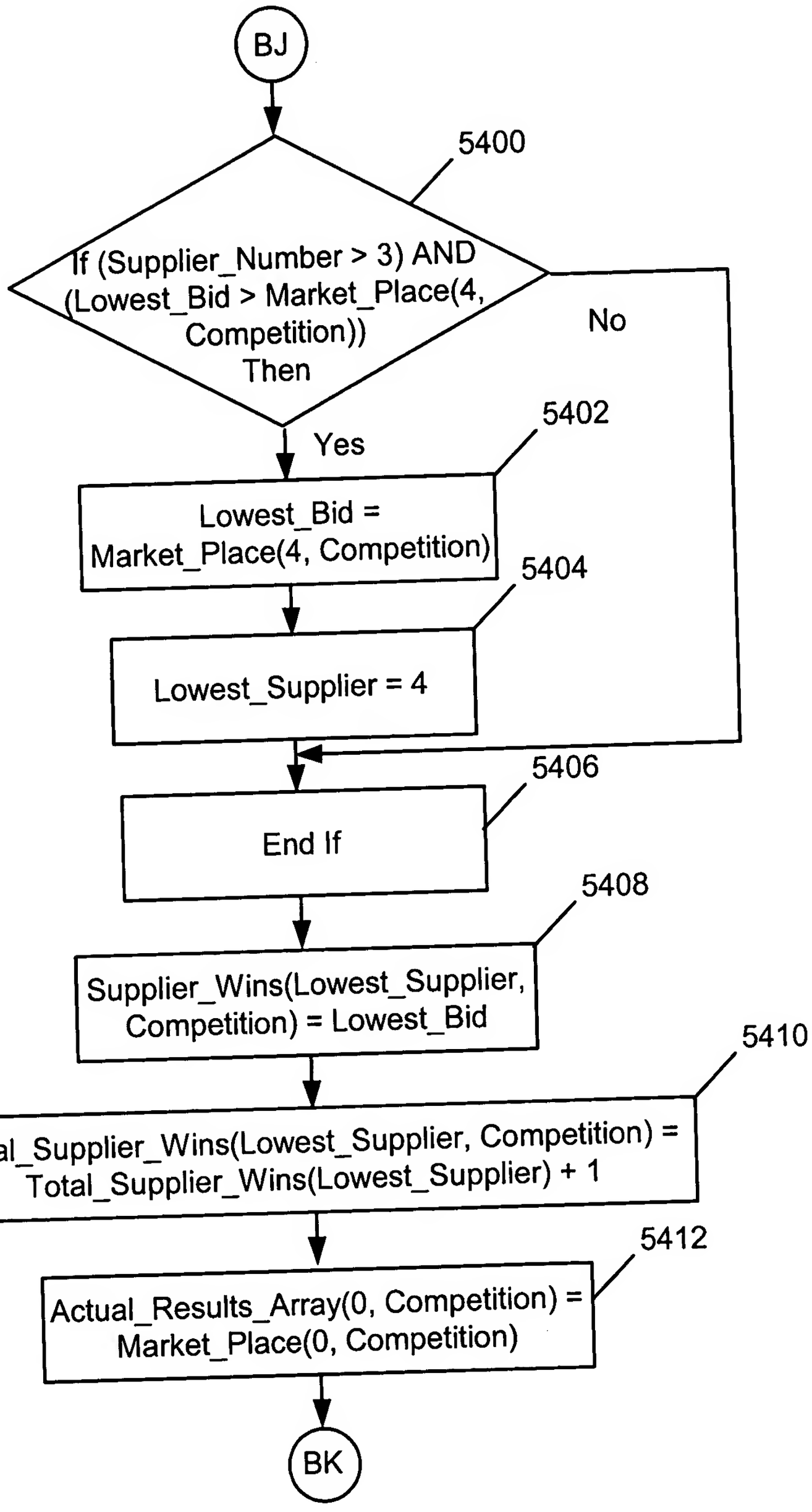

BJ
5400
If (Supplier_Number > 3) AND (Lowest_Bid > Market_Place(4, Competition)) Then
No
Yes
5402
Lowest_Bid = Market_Place(4, Competition)
5404
Lowest_Supplier = 4
5406
End If
5408
Supplier_Wins(Lowest_Supplier, Competition) = Lowest_Bid
5410
Total_Supplier_Wins(Lowest_Supplier, Competition) = Total_Supplier_Wins(Lowest_Supplier) + 1
5412
Actual_Results_Array(0, Competition) = Market_Place(0, Competition)
BK

Fig 55

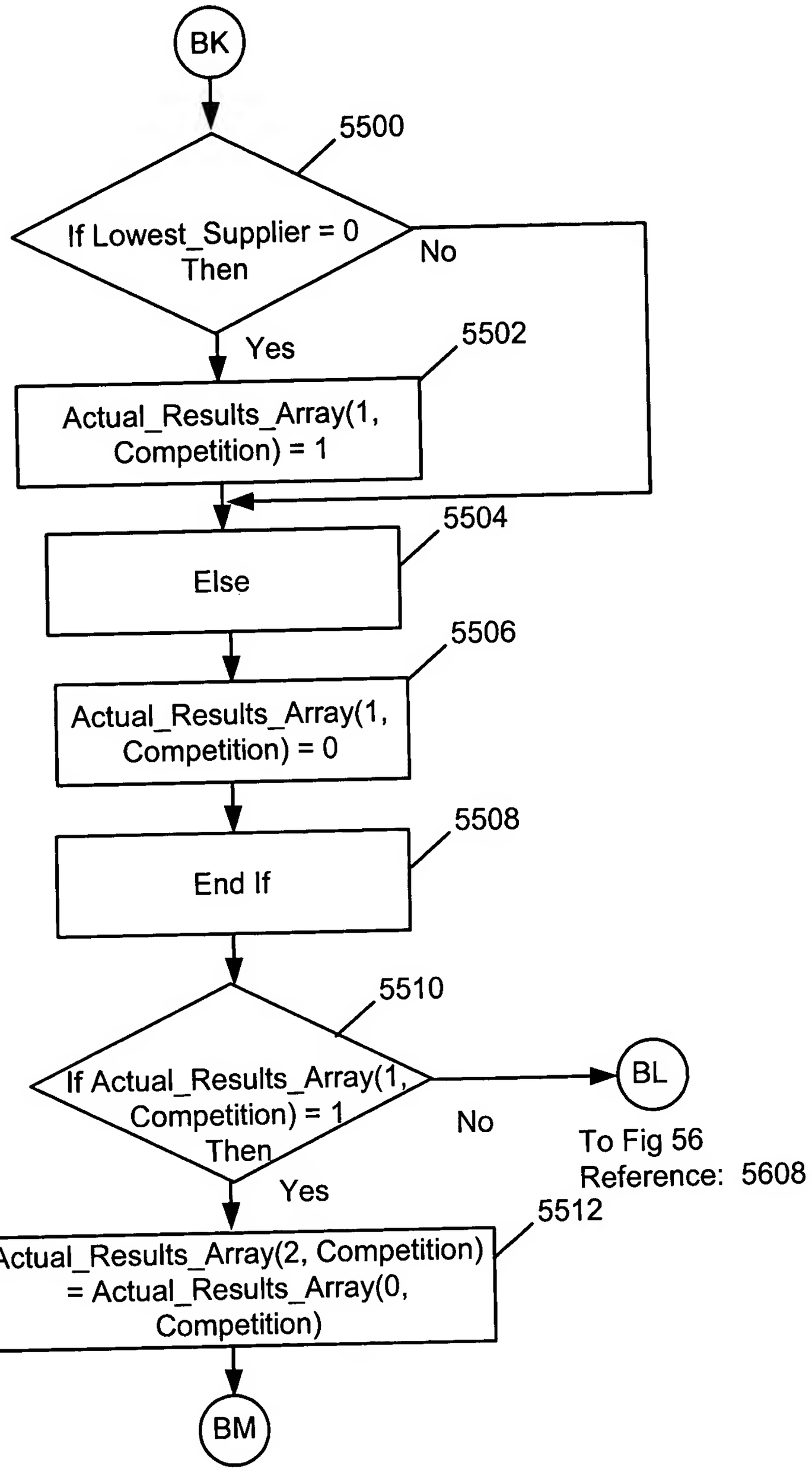

BK
5500
If Lowest_Supplier = 0 Then
No
Yes
5502
Actual_Results_Array(1, Competition) = 1
5504
Else
5506
Actual_Results_Array(1, Competition) = 0
5508
End If
5510
If Actual_Results_Array(1, Competition) = 1 Then
No
BL
To Fig 56
Reference: 5608
Yes
5512
Actual_Results_Array(2, Competition) = Actual_Results_Array(0, Competition)
BM

Fig 56

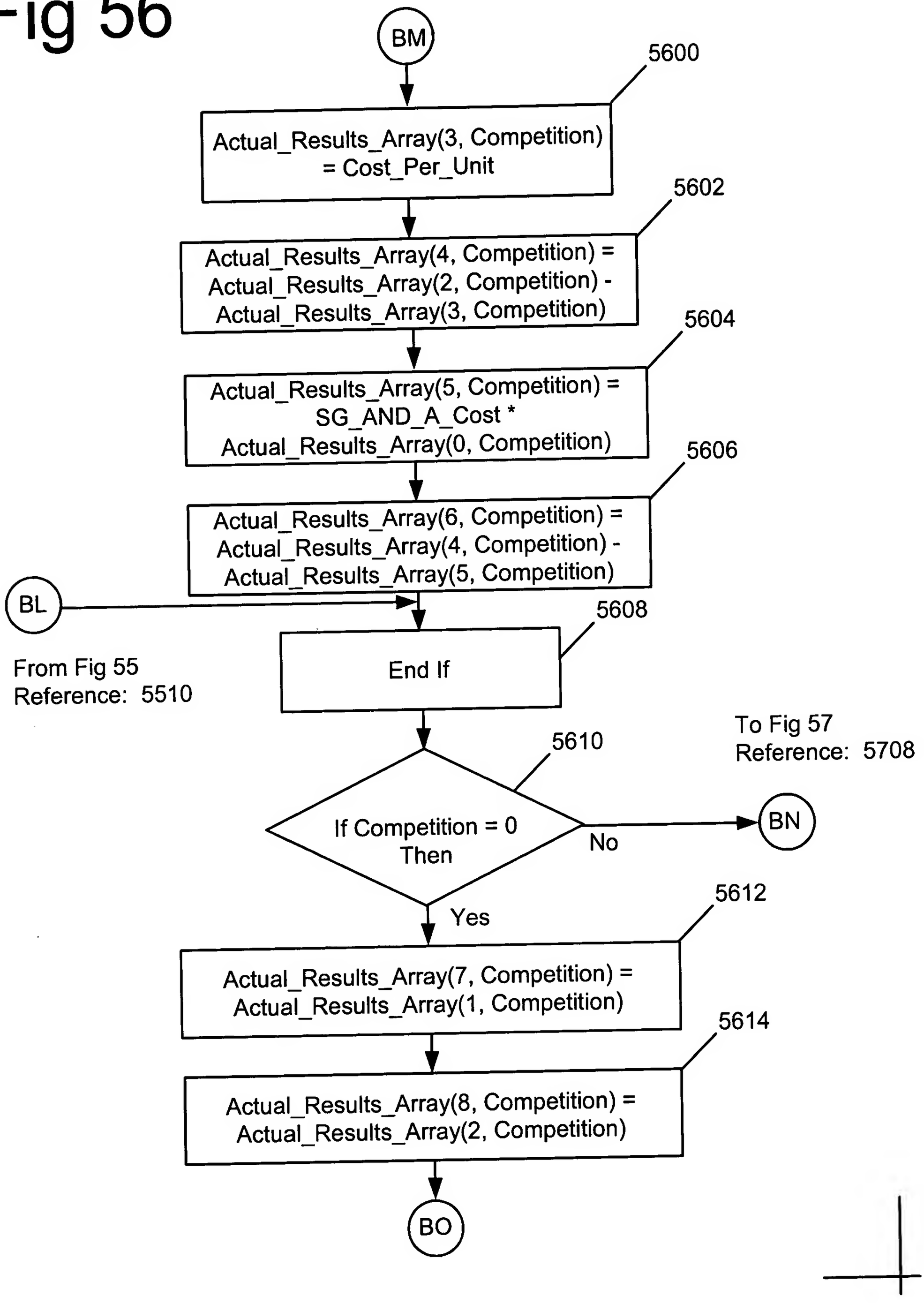

BM
5600
Actual_Results_Array(3, Competition) = Cost_Per_Unit
5602
Actual_Results_Array(4, Competition) = Actual_Results_Array(2, Competition) - Actual_Results_Array(3, Competition)
5604
Actual_Results_Array(5, Competition) = SG_AND_A_Cost * Actual_Results_Array(0, Competition)
5606
Actual_Results_Array(6, Competition) = Actual_Results_Array(4, Competition) - Actual_Results_Array(5, Competition)
BL
From Fig 55
Reference: 5510
5608
End If
5610
If Competition = 0 Then
No
BN
To Fig 57
Reference: 5708
Yes
5612
Actual_Results_Array(7, Competition) = Actual_Results_Array(1, Competition)
5614
Actual_Results_Array(8, Competition) = Actual_Results_Array(2, Competition)
BO

Fig 57

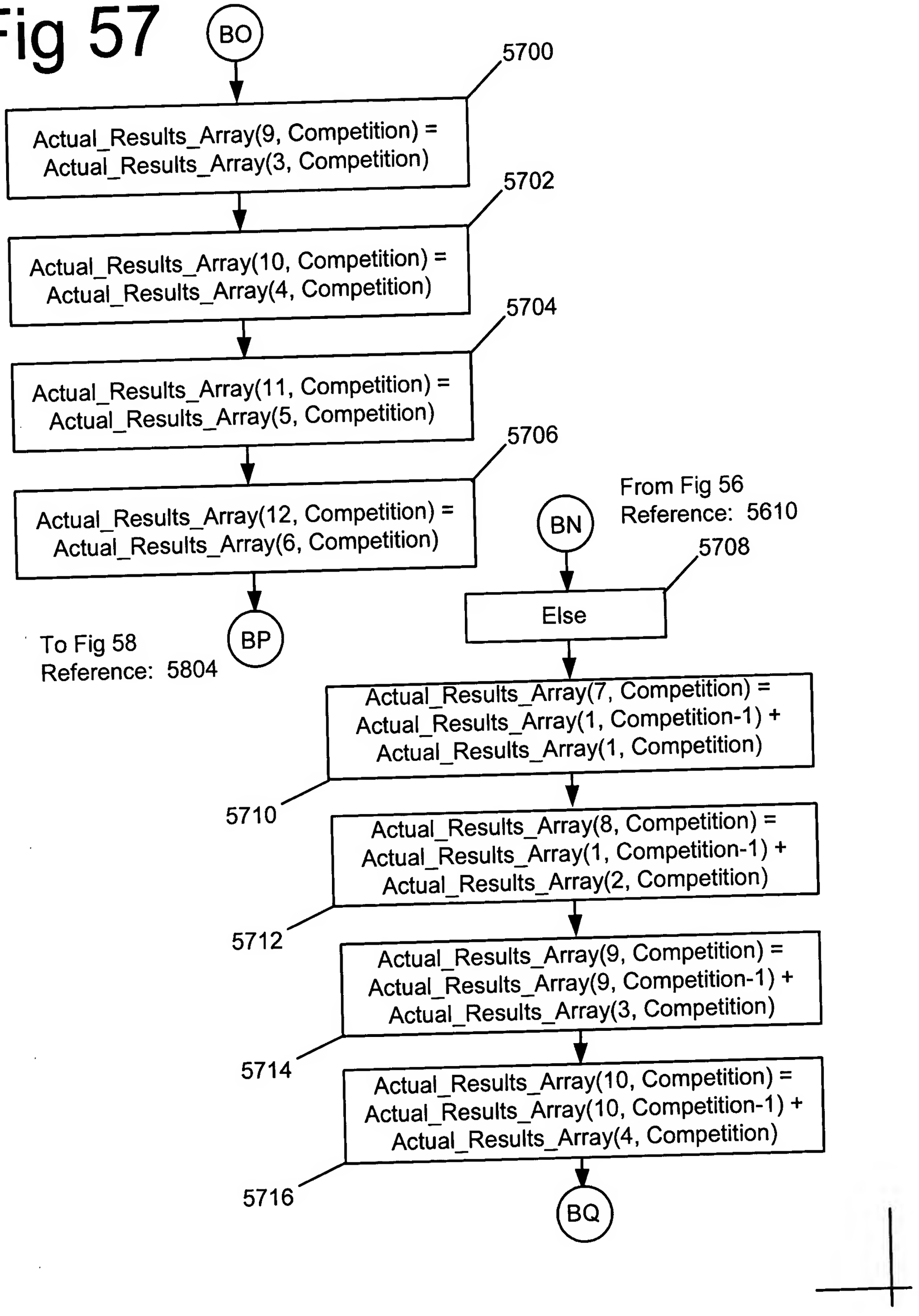

BO
5700
Actual_Results_Array(9, Competition) =
Actual_Results_Array(3, Competition)
5702
Actual_Results_Array(10, Competition) =
Actual_Results_Array(4, Competition)
5704
Actual_Results_Array(11, Competition) =
Actual_Results_Array(5, Competition)
5706
Actual_Results_Array(12, Competition) =
Actual_Results_Array(6, Competition)
BP
To Fig 58
Reference: 5804
BN
From Fig 56
Reference: 5610
5708
Else
Actual_Results_Array(7, Competition) =
Actual_Results_Array(1, Competition-1) +
Actual_Results_Array(1, Competition)
5710
Actual_Results_Array(8, Competition) =
Actual_Results_Array(1, Competition-1) +
Actual_Results_Array(2, Competition)
5712
Actual_Results_Array(9, Competition) =
Actual_Results_Array(9, Competition-1) +
Actual_Results_Array(3, Competition)
5714
Actual_Results_Array(10, Competition) =
Actual_Results_Array(10, Competition-1) +
Actual_Results_Array(4, Competition)
5716
BQ

Fig 58

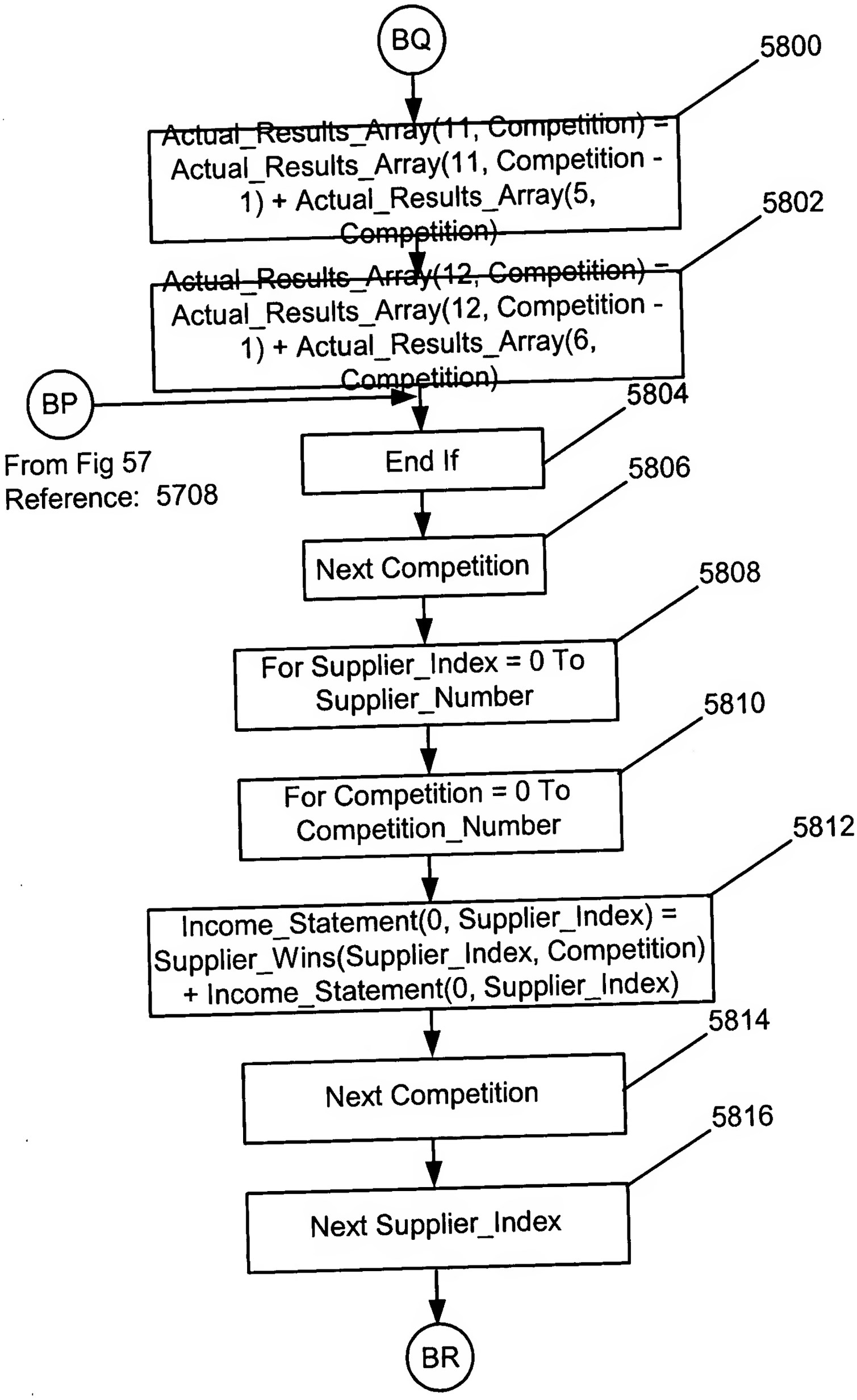

BQ
5800
Actual_Results_Array(11, Competition) = Actual_Results_Array(11, Competition - 1) + Actual_Results_Array(5, Competition)
5802
Actual_Results_Array(12, Competition) = Actual_Results_Array(12, Competition - 1) + Actual_Results_Array(6, Competition)
BP
From Fig 57
Reference: 5708
5804
End If
5806
Next Competition
5808
For Supplier_Index = 0 To Supplier_Number
5810
For Competition = 0 To Competition_Number
5812
Income_Statement(0, Supplier_Index) = Supplier_Wins(Supplier_Index, Competition) + Income_Statement(0, Supplier_Index)
5814
Next Competition
5816
Next Supplier_Index
BR

Fig 59

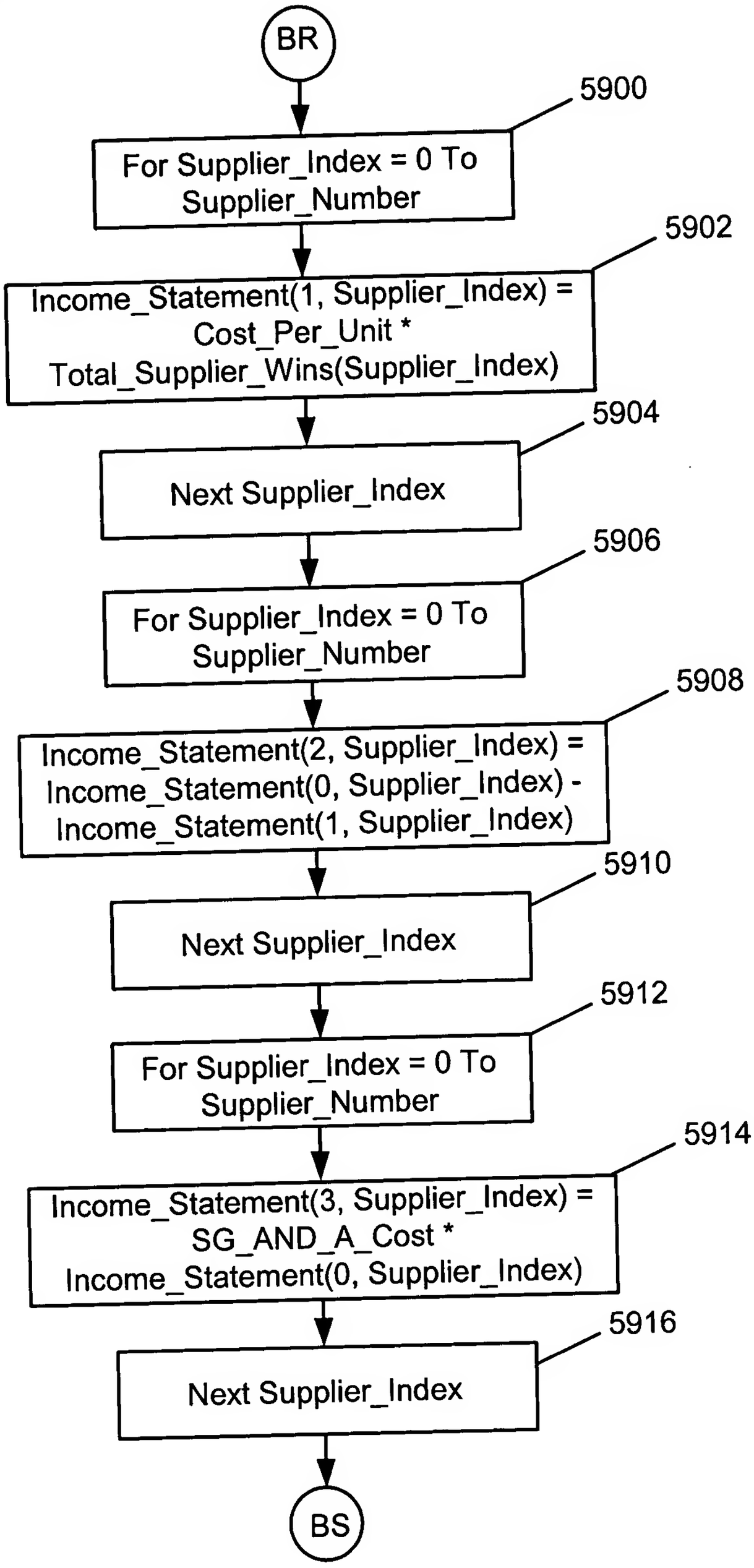

BR
5900
For Supplier_Index = 0 To Supplier_Number
5902
Income_Statement(1, Supplier_Index) = Cost_Per_Unit * Total_Supplier_Wins(Supplier_Index)
5904
Next Supplier_Index
5906
For Supplier_Index = 0 To Supplier_Number
5908
Income_Statement(2, Supplier_Index) = Income_Statement(0, Supplier_Index) - Income_Statement(1, Supplier_Index)
5910
Next Supplier_Index
5912
For Supplier_Index = 0 To Supplier_Number
5914
Income_Statement(3, Supplier_Index) = SG_AND_A_Cost * Income_Statement(0, Supplier_Index)
5916
Next Supplier_Index
BS

Fig 60

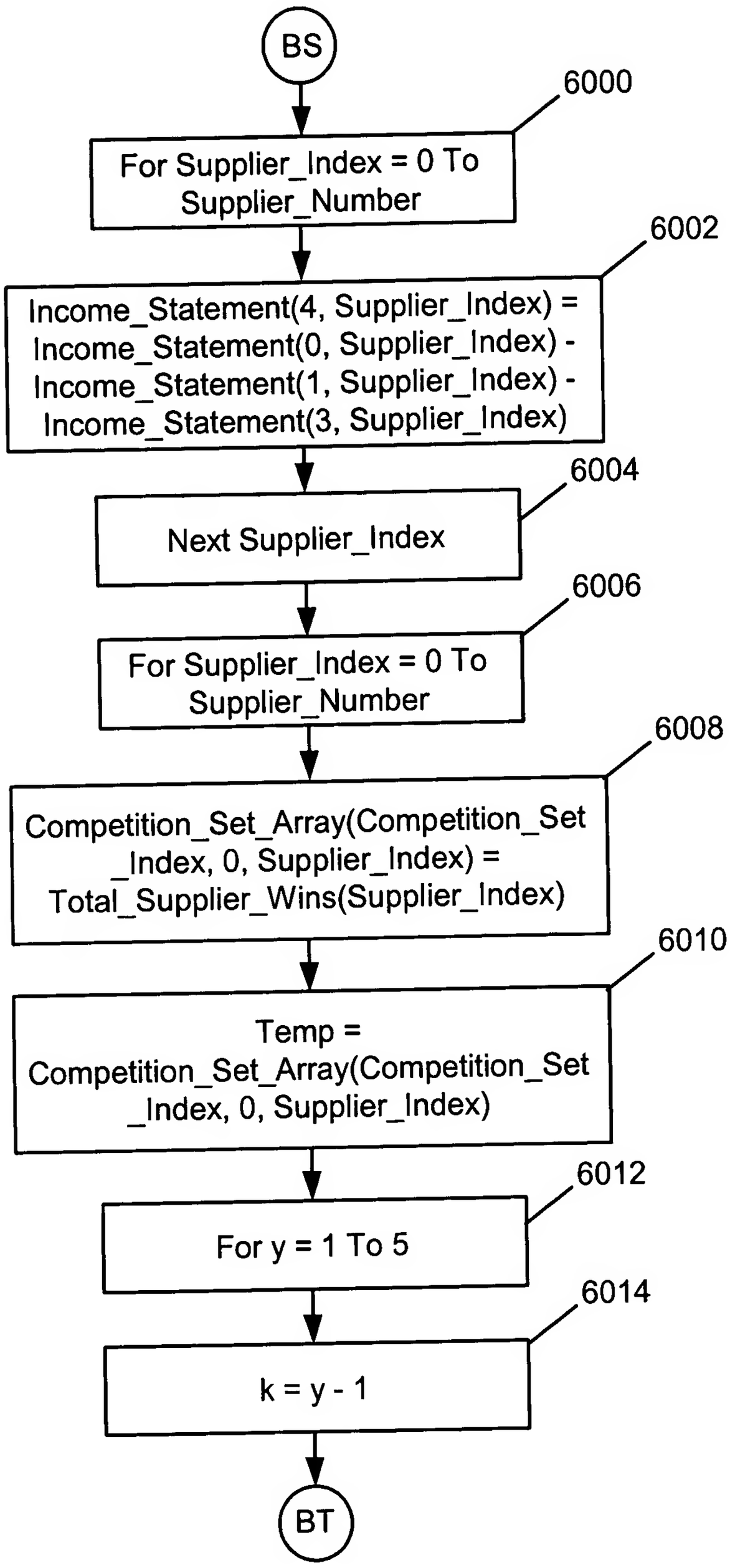

BS
6000
For Supplier_Index = 0 To Supplier_Number
6002
Income_Statement(4, Supplier_Index) = Income_Statement(0, Supplier_Index) - Income_Statement(1, Supplier_Index) - Income_Statement(3, Supplier_Index)
6004
Next Supplier_Index
6006
For Supplier_Index = 0 To Supplier_Number
6008
Competition_Set_Array(Competition_Set_Index, 0, Supplier_Index) = Total_Supplier_Wins(Supplier_Index)
6010
Temp = Competition_Set_Array(Competition_Set_Index, 0, Supplier_Index)
6012
For y = 1 To 5
6014
k = y - 1
BT

Fig 61

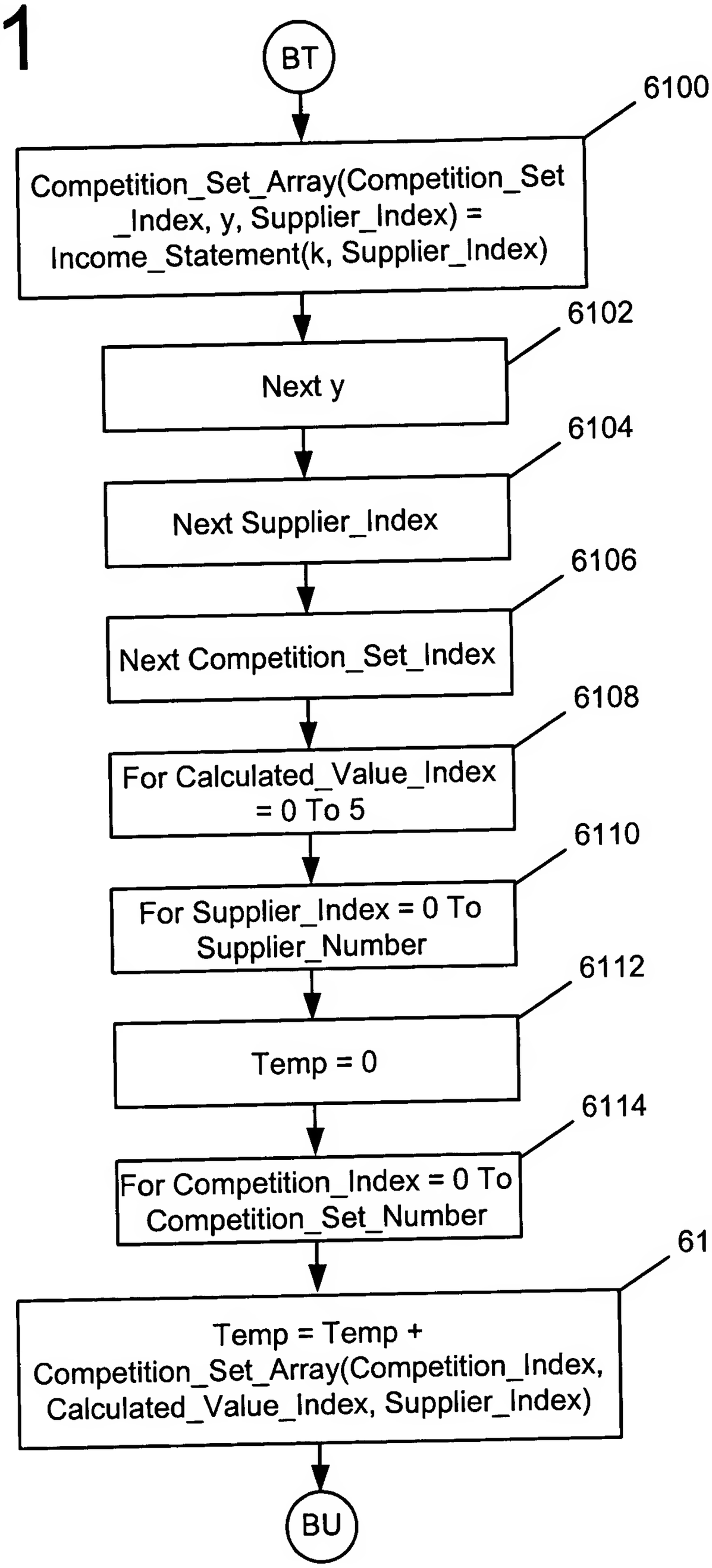

BT
6100
Competition_Set_Array(Competition_Set_Index, y, Supplier_Index) = Income_Statement(k, Supplier_Index)
6102
Next y
6104
Next Supplier_Index
6106
Next Competition_Set_Index
6108
For Calculated_Value_Index = 0 To 5
6110
For Supplier_Index = 0 To Supplier_Number
6112
Temp = 0
6114
For Competition_Index = 0 To Competition_Set_Number
6116
Temp = Temp + Competition_Set_Array(Competition_Index, Calculated_Value_Index, Supplier_Index)
BU

Fig 62

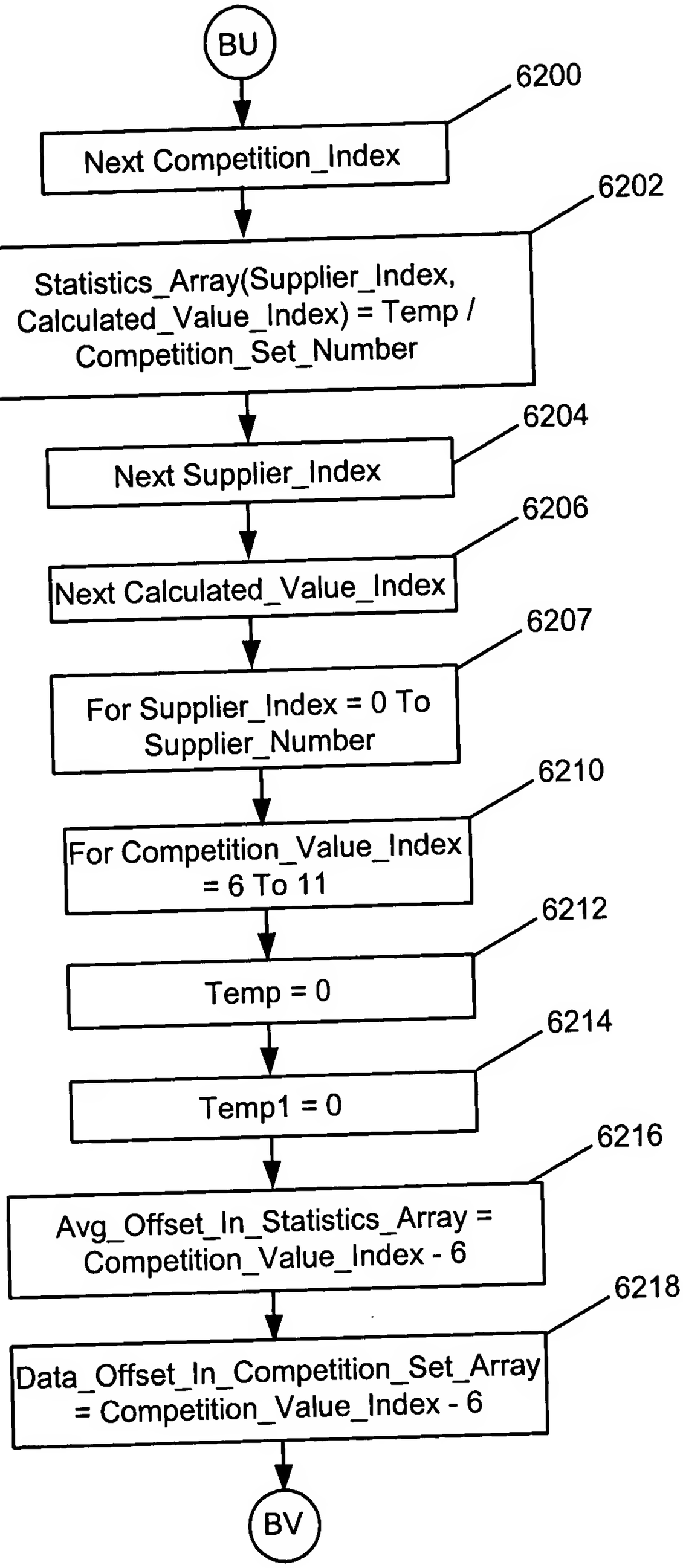

BU
6200
Next Competition_Index
6202
Statistics_Array(Supplier_Index, Calculated_Value_Index) = Temp / Competition_Set_Number
6204
Next Supplier_Index
6206
Next Calculated_Value_Index
6207
For Supplier_Index = 0 To Supplier_Number
6210
For Competition_Value_Index = 6 To 11
6212
Temp = 0
6214
Temp1 = 0
6216
Avg_Offset_In_Statistics_Array = Competition_Value_Index - 6
6218
Data_Offset_In_Competition_Set_Array = Competition_Value_Index - 6
BV

Fig 63

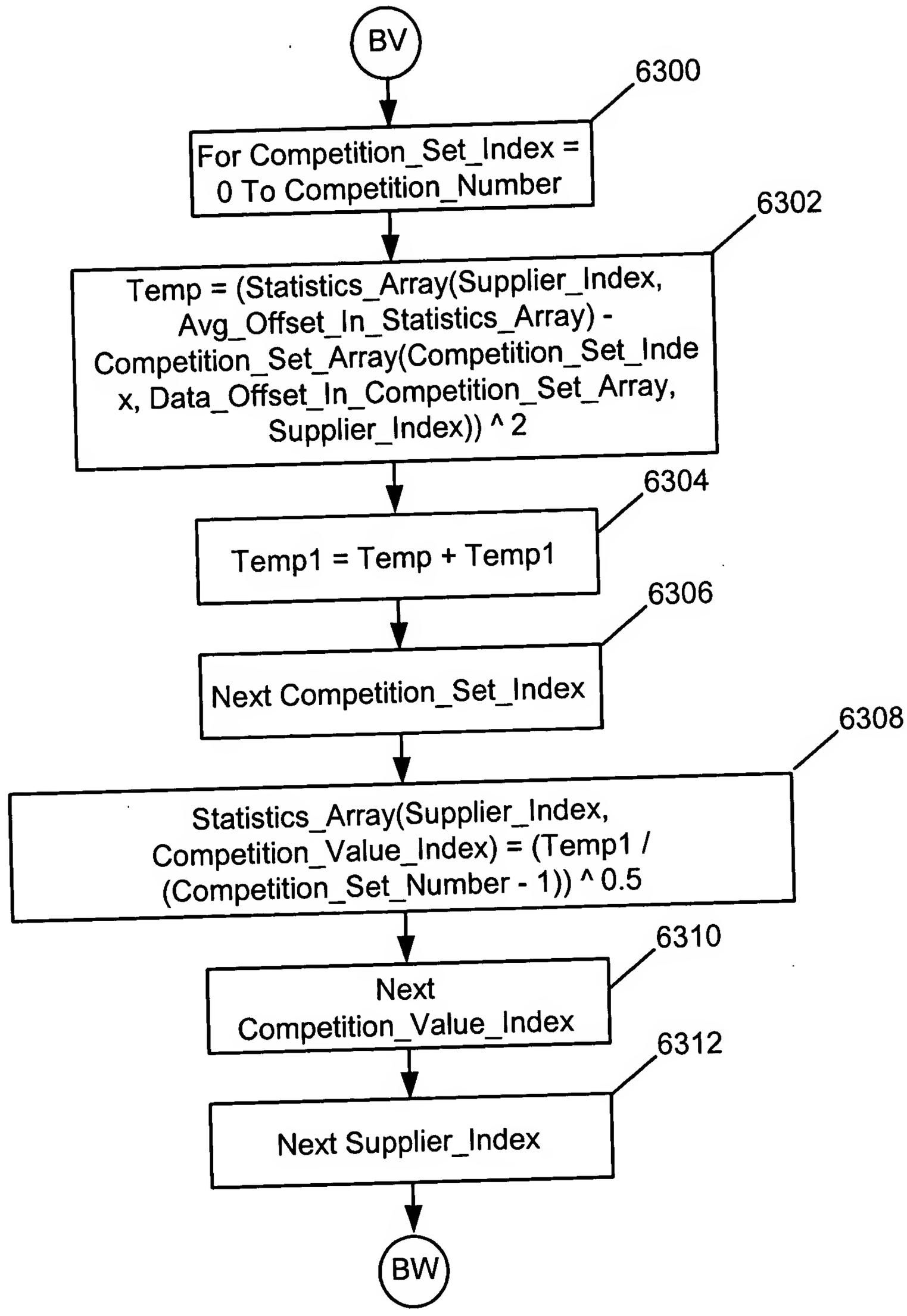

BV
6300
For Competition_Set_Index = 0 To Competition_Number
6302
Temp = (Statistics_Array(Supplier_Index, Avg_Offset_In_Statistics_Array) - Competition_Set_Array(Competition_Set_Index, Data_Offset_In_Competition_Set_Array, Supplier_Index)) ^ 2
6304
Temp1 = Temp + Temp1
6306
Next Competition_Set_Index
6308
Statistics_Array(Supplier_Index, Competition_Value_Index) = (Temp1 / (Competition_Set_Number - 1)) ^ 0.5
6310
Next Competition_Value_Index
6312
Next Supplier_Index
BW

Fig 64

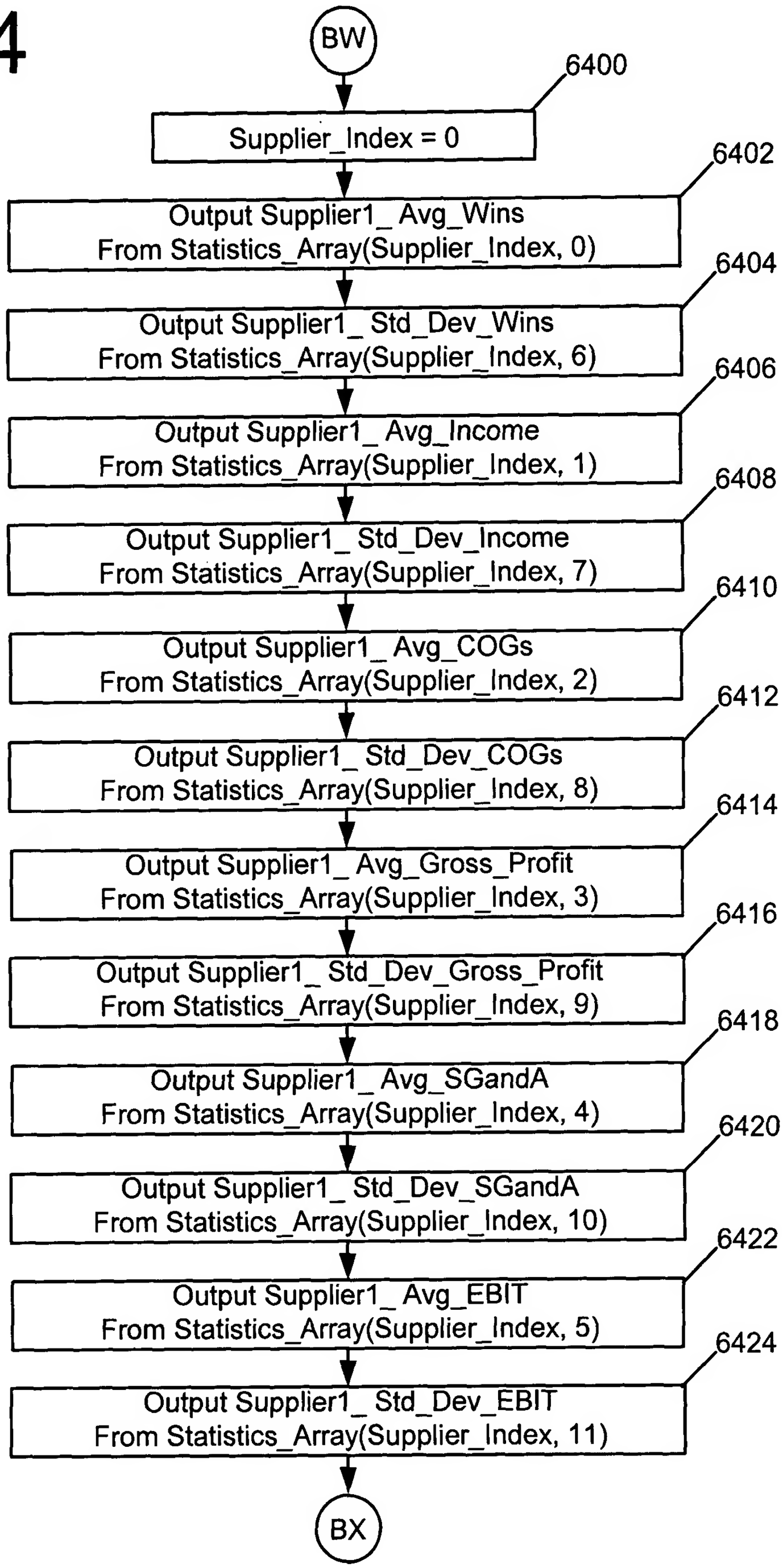

BW
6400
Supplier_Index = 0
6402
Output Supplier1_ Avg_Wins
From Statistics_Array(Supplier_Index, 0)
6404
Output Supplier1_ Std_Dev_Wins
From Statistics_Array(Supplier_Index, 6)
6406
Output Supplier1_ Avg_Income
From Statistics_Array(Supplier_Index, 1)
6408
Output Supplier1_ Std_Dev_Income
From Statistics_Array(Supplier_Index, 7)
6410
Output Supplier1_ Avg_COGs
From Statistics_Array(Supplier_Index, 2)
6412
Output Supplier1_ Std_Dev_COGs
From Statistics_Array(Supplier_Index, 8)
6414
Output Supplier1_ Avg_Gross_Profit
From Statistics_Array(Supplier_Index, 3)
6416
Output Supplier1_ Std_Dev_Gross_Profit
From Statistics_Array(Supplier_Index, 9)
6418
Output Supplier1_ Avg_SGandA
From Statistics_Array(Supplier_Index, 4)
6420
Output Supplier1_ Std_Dev_SGandA
From Statistics_Array(Supplier_Index, 10)
6422
Output Supplier1_ Avg_EBIT
From Statistics_Array(Supplier_Index, 5)
6424
Output Supplier1_ Std_Dev_EBIT
From Statistics_Array(Supplier_Index, 11)
BX

Fig 65

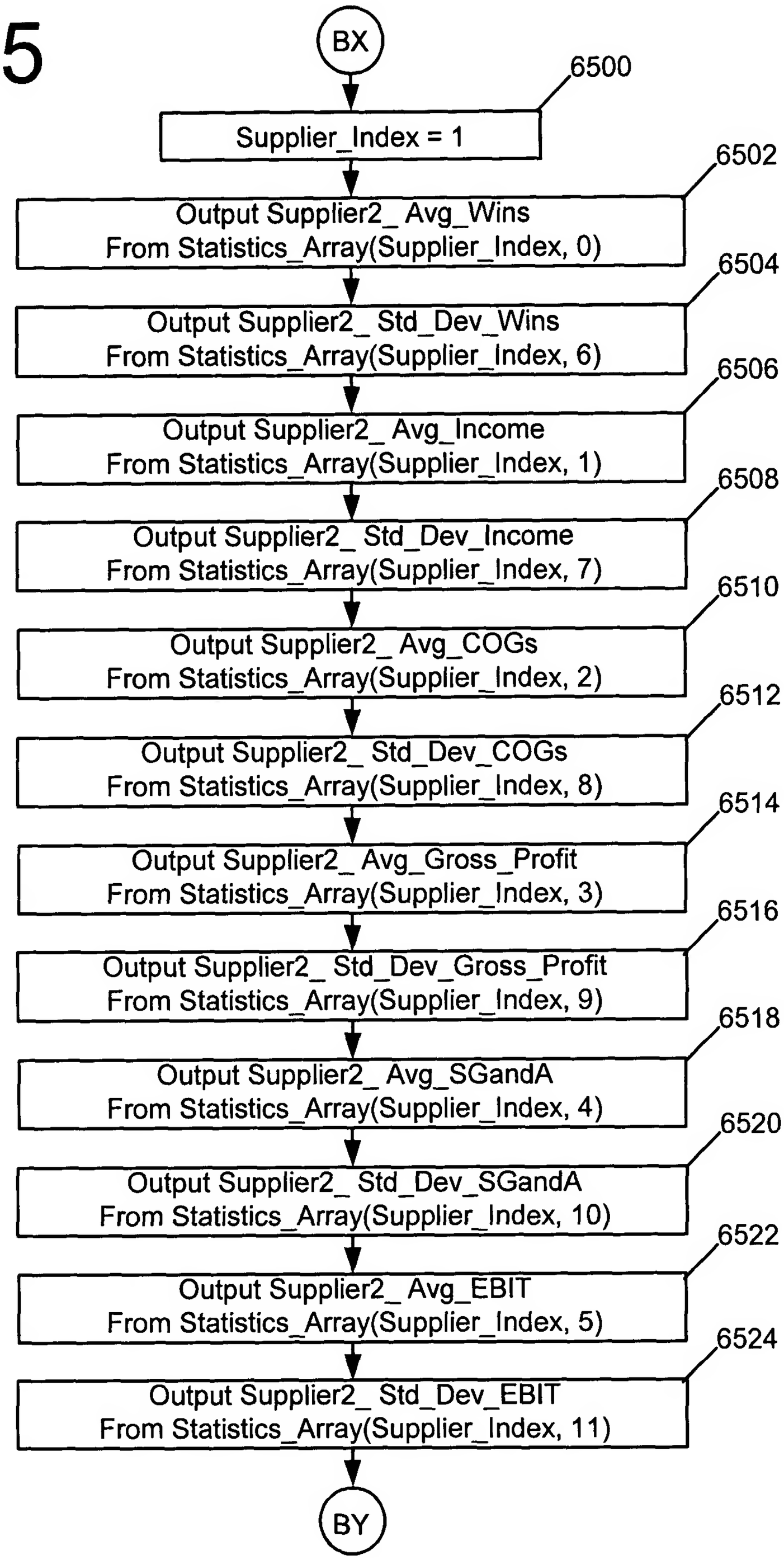

BX
6500
Supplier_Index = 1
6502
Output Supplier2_ Avg_Wins
From Statistics_Array(Supplier_Index, 0)
6504
Output Supplier2_ Std_Dev_Wins
From Statistics_Array(Supplier_Index, 6)
6506
Output Supplier2_ Avg_Income
From Statistics_Array(Supplier_Index, 1)
6508
Output Supplier2_ Std_Dev_Income
From Statistics_Array(Supplier_Index, 7)
6510
Output Supplier2_ Avg_COGs
From Statistics_Array(Supplier_Index, 2)
6512
Output Supplier2_ Std_Dev_COGs
From Statistics_Array(Supplier_Index, 8)
6514
Output Supplier2_ Avg_Gross_Profit
From Statistics_Array(Supplier_Index, 3)
6516
Output Supplier2_ Std_Dev_Gross_Profit
From Statistics_Array(Supplier_Index, 9)
6518
Output Supplier2_ Avg_SGandA
From Statistics_Array(Supplier_Index, 4)
6520
Output Supplier2_ Std_Dev_SGandA
From Statistics_Array(Supplier_Index, 10)
6522
Output Supplier2_ Avg_EBIT
From Statistics_Array(Supplier_Index, 5)
6524
Output Supplier2_ Std_Dev_EBIT
From Statistics_Array(Supplier_Index, 11)
BY

Fig 66

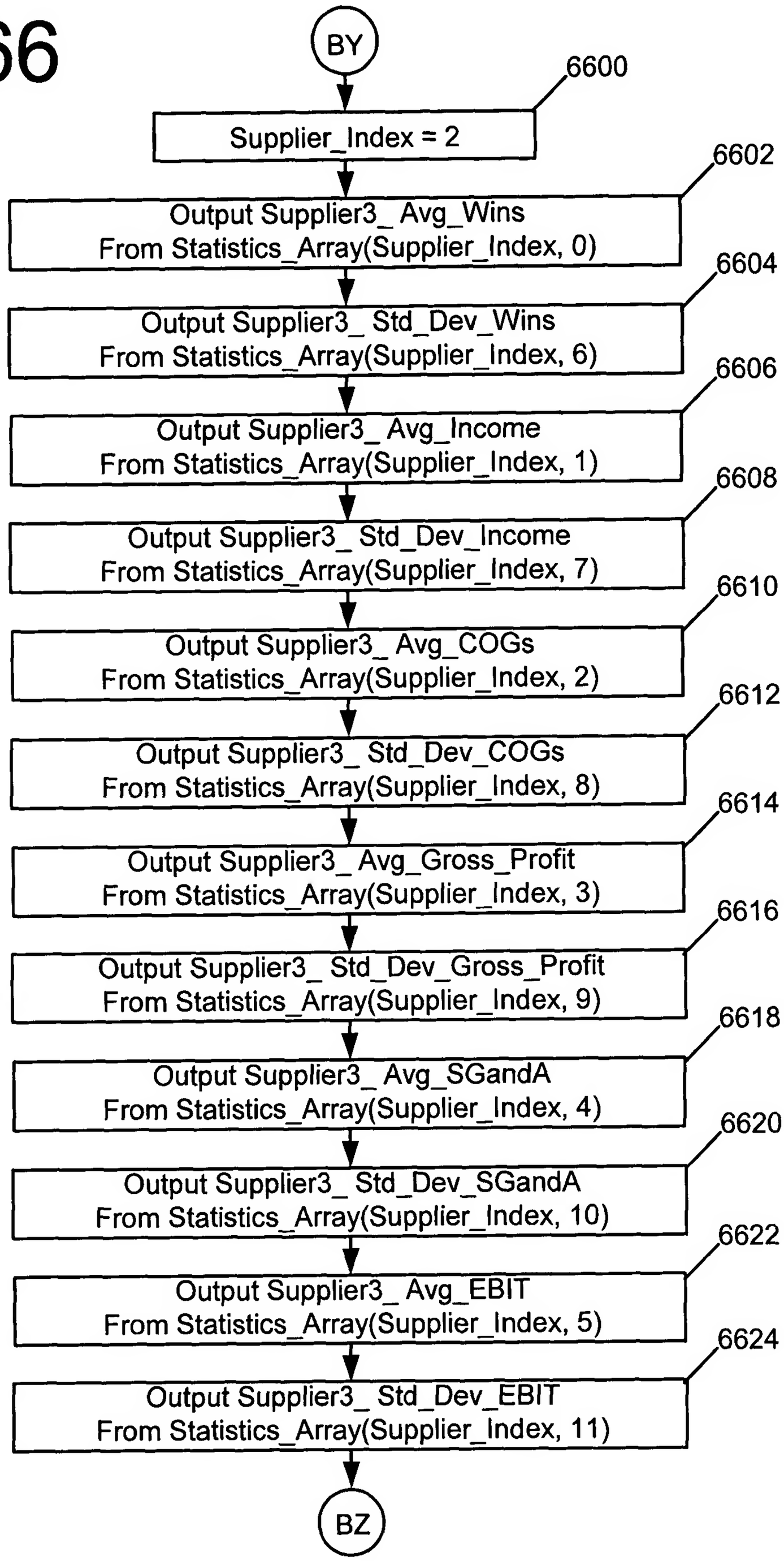

BY
6600
Supplier_Index = 2
6602
Output Supplier3_ Avg_Wins
From Statistics_Array(Supplier_Index, 0)
6604
Output Supplier3_ Std_Dev_Wins
From Statistics_Array(Supplier_Index, 6)
6606
Output Supplier3_ Avg_Income
From Statistics_Array(Supplier_Index, 1)
6608
Output Supplier3_ Std_Dev_Income
From Statistics_Array(Supplier_Index, 7)
6610
Output Supplier3_ Avg_COGs
From Statistics_Array(Supplier_Index, 2)
6612
Output Supplier3_ Std_Dev_COGs
From Statistics_Array(Supplier_Index, 8)
6614
Output Supplier3_ Avg_Gross_Profit
From Statistics_Array(Supplier_Index, 3)
6616
Output Supplier3_ Std_Dev_Gross_Profit
From Statistics_Array(Supplier_Index, 9)
6618
Output Supplier3_ Avg_SGandA
From Statistics_Array(Supplier_Index, 4)
6620
Output Supplier3_ Std_Dev_SGandA
From Statistics_Array(Supplier_Index, 10)
6622
Output Supplier3_ Avg_EBIT
From Statistics_Array(Supplier_Index, 5)
6624
Output Supplier3_ Std_Dev_EBIT
From Statistics_Array(Supplier_Index, 11)
BZ

Fig 67

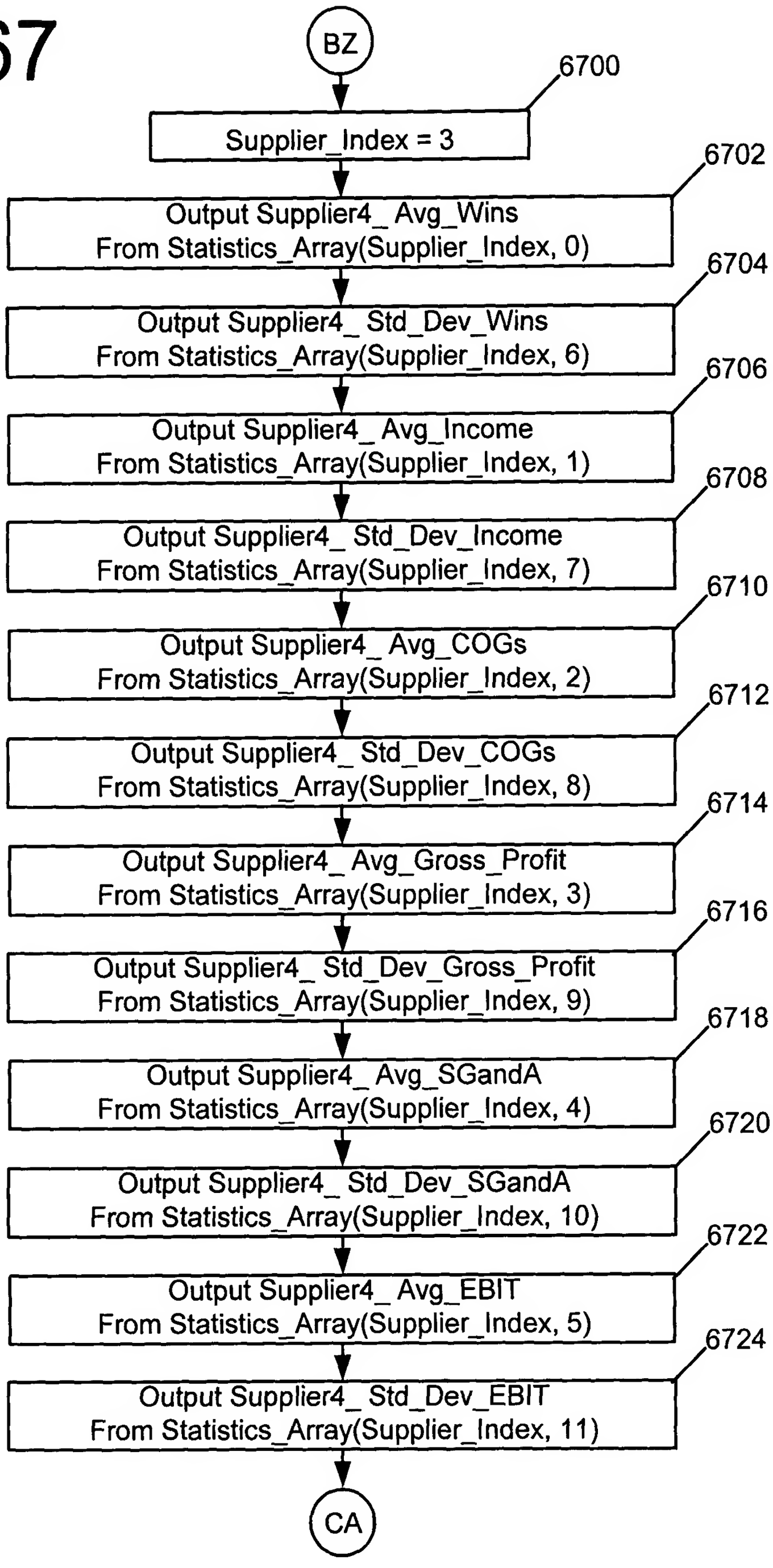

BZ
6700
Supplier_Index = 3
6702
Output Supplier4_ Avg_Wins
From Statistics_Array(Supplier_Index, 0)
6704
Output Supplier4_ Std_Dev_Wins
From Statistics_Array(Supplier_Index, 6)
6706
Output Supplier4_ Avg_Income
From Statistics_Array(Supplier_Index, 1)
6708
Output Supplier4_ Std_Dev_Income
From Statistics_Array(Supplier_Index, 7)
6710
Output Supplier4_ Avg_COGs
From Statistics_Array(Supplier_Index, 2)
6712
Output Supplier4_ Std_Dev_COGs
From Statistics_Array(Supplier_Index, 8)
6714
Output Supplier4_ Avg_Gross_Profit
From Statistics_Array(Supplier_Index, 3)
6716
Output Supplier4_ Std_Dev_Gross_Profit
From Statistics_Array(Supplier_Index, 9)
6718
Output Supplier4_ Avg_SGandA
From Statistics_Array(Supplier_Index, 4)
6720
Output Supplier4_ Std_Dev_SGandA
From Statistics_Array(Supplier_Index, 10)
6722
Output Supplier4_ Avg_EBIT
From Statistics_Array(Supplier_Index, 5)
6724
Output Supplier4_ Std_Dev_EBIT
From Statistics_Array(Supplier_Index, 11)
CA

Fig 68

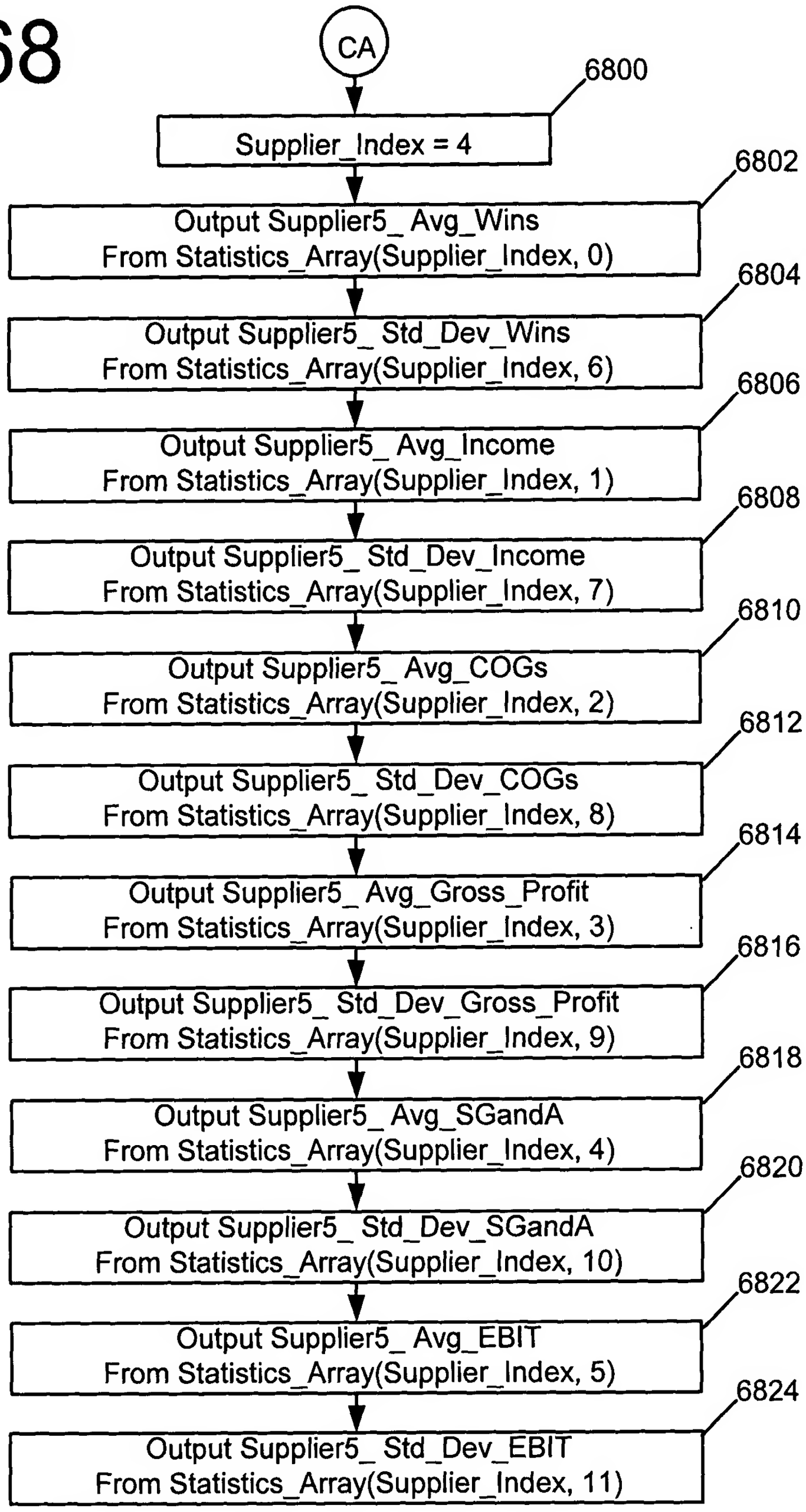

CA
6800
Supplier_Index = 4
6802
Output Supplier5_ Avg_Wins
From Statistics_Array(Supplier_Index, 0)
6804
Output Supplier5_ Std_Dev_Wins
From Statistics_Array(Supplier_Index, 6)
6806
Output Supplier5_ Avg_Income
From Statistics_Array(Supplier_Index, 1)
6808
Output Supplier5_ Std_Dev_Income
From Statistics_Array(Supplier_Index, 7)
6810
Output Supplier5_ Avg_COGs
From Statistics_Array(Supplier_Index, 2)
6812
Output Supplier5_ Std_Dev_COGs
From Statistics_Array(Supplier_Index, 8)
6814
Output Supplier5_ Avg_Gross_Profit
From Statistics_Array(Supplier_Index, 3)
6816
Output Supplier5_ Std_Dev_Gross_Profit
From Statistics_Array(Supplier_Index, 9)
6818
Output Supplier5_ Avg_SGandA
From Statistics_Array(Supplier_Index, 4)
6820
Output Supplier5_ Std_Dev_SGandA
From Statistics_Array(Supplier_Index, 10)
6822
Output Supplier5_ Avg_EBIT
From Statistics_Array(Supplier_Index, 5)
6824
Output Supplier5_ Std_Dev_EBIT
From Statistics_Array(Supplier_Index, 11)

Fig 69

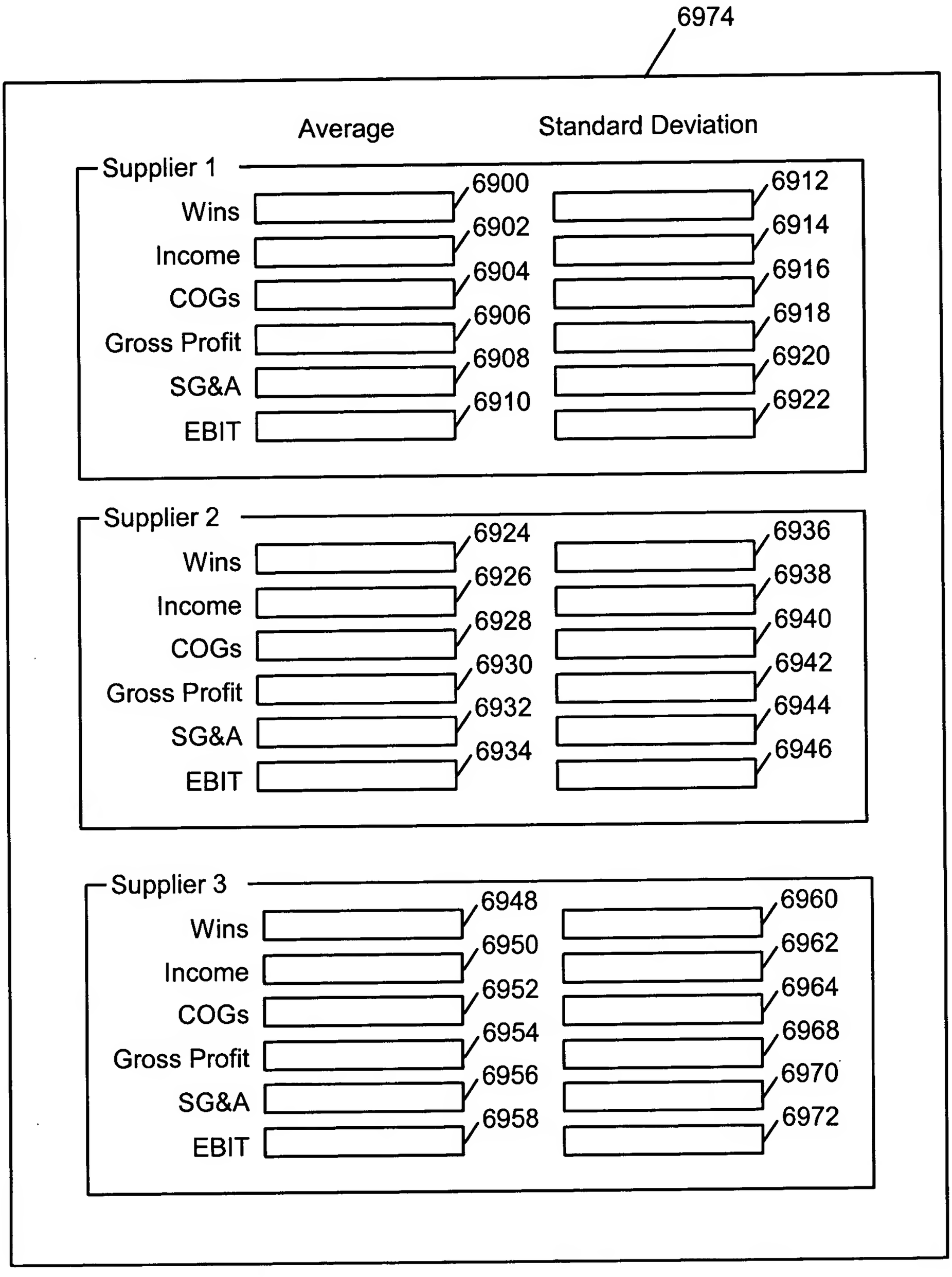
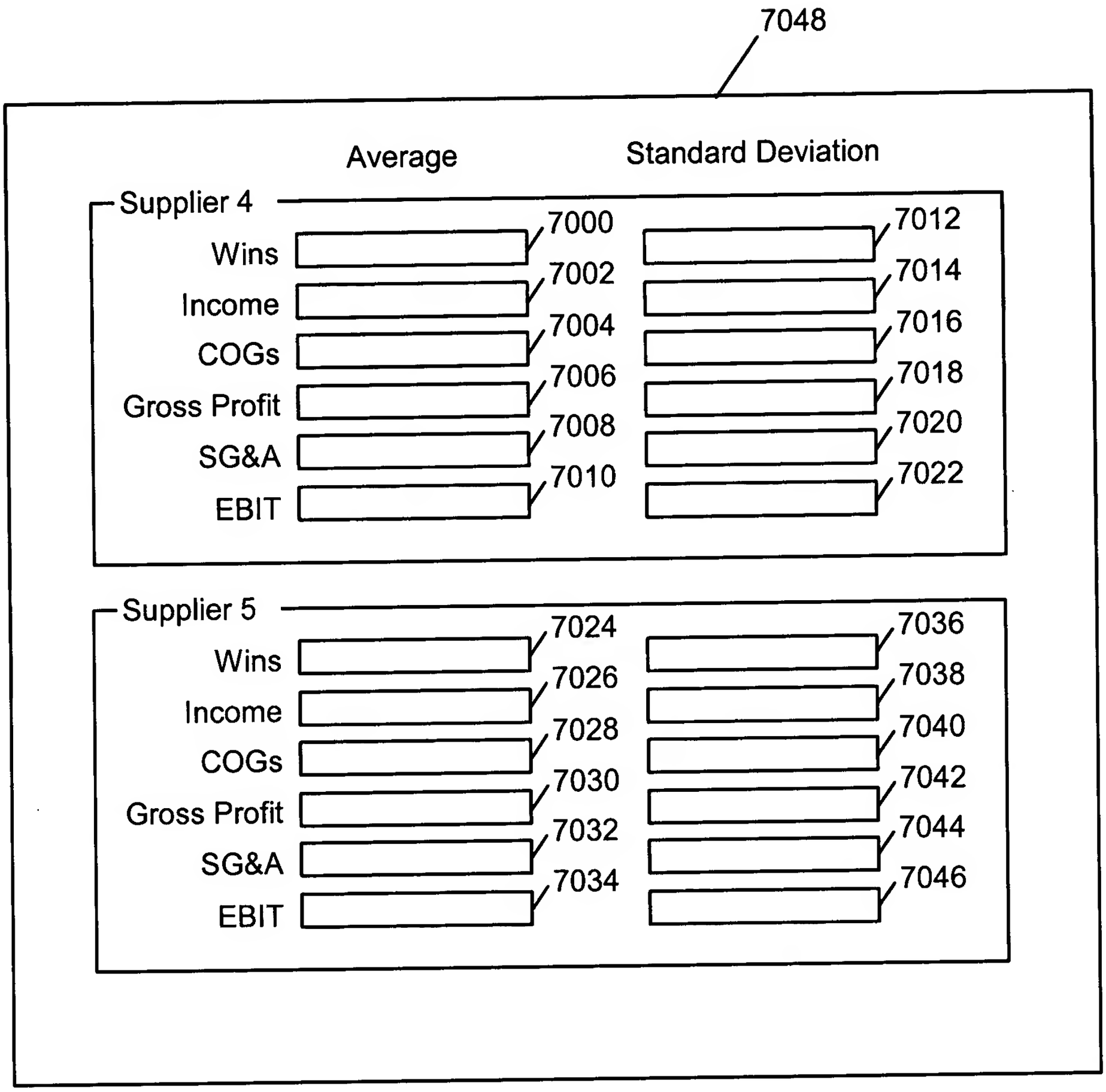

6974
Average
Standard Deviation
Supplier 1
Wins
Income
COGs
Gross Profit
SG&A
EBIT
6900
6902
6904
6906
6908
6910
6912
6914
6916
6918
6920
6922
Supplier 2
Wins
Income
COGs
Gross Profit
SG&A
EBIT
6924
6926
6928
6930
6932
6934
6936
6938
6940
6942
6944
6946
Supplier 3
Wins
Income
COGs
Gross Profit
SG&A
EBIT
6948
6950
6952
6954
6956
6958
6960
6962
6964
6968
6970
6972

Fig 70


7048
Average
Standard Deviation
Supplier 4
Wins
Income
COGs
Gross Profit
SG&A
EBIT
7000
7002
7004
7006
7008
7010
7012
7014
7016
7018
7020
7022
Supplier 5
Wins
Income
COGs
Gross Profit
SG&A
EBIT
7024
7026
7028
7030
7032
7034
7036
7038
7040
7042
7044
7046